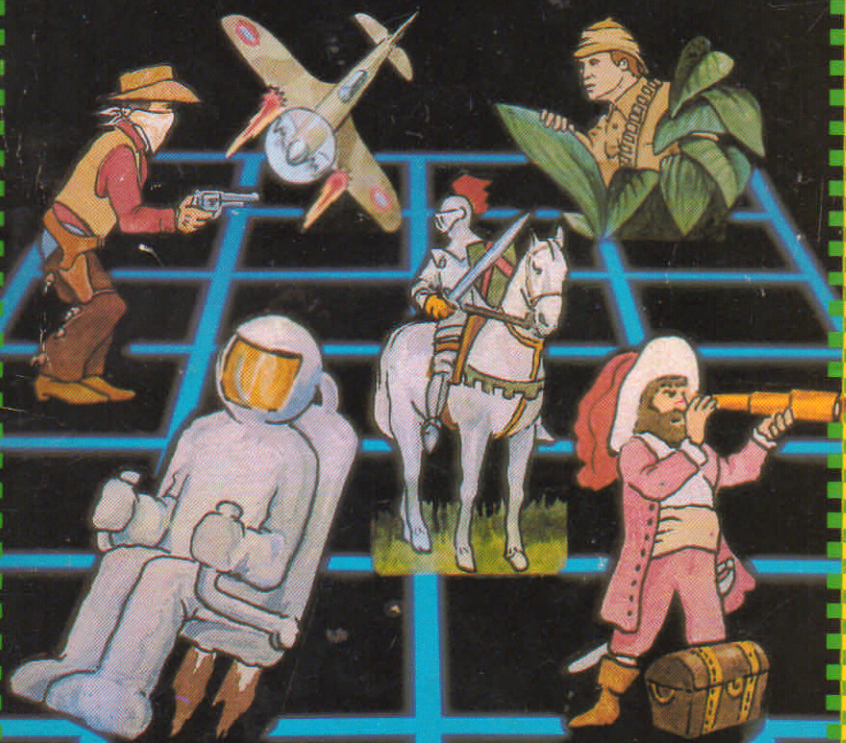


THE VIRGIN COMPUTER SERIES  
Series editors: Tim Hartnell & Clive Gifford

# ADVENTURES FOR YOUR ZX SPECTRUM

CLIVE GIFFORD



£££££'s of entertaining games for only £2.95



**ADVENTURES  
FOR YOUR  
ZX  
SPECTRUM**

**By  
Clive Gifford**

*Virgin*

Virgin Books

First published in Great Britain in 1984 by Virgin Books Ltd,  
61-63 Portobello Road, London W11 3DD.

Copyright © 1984 Interface/Virgin Books

ISBN 0 86369 060 2

All rights reserved. No part of this book may be reproduced in  
any form or by any means without prior permission from the  
publisher.

Printed and bound in Great Britain by Richard Clay  
(The Chaucer Press) Ltd, Suffolk.

Production services by Book Production Consultants,  
Cambridge.

Designed by Ray Hyden.

Illustrated by Sue Walliker.

Typeset by QV Typesetting.

Distributed by Arrow Books.

### **TIM HARTNELL — SERIES EDITOR**

Tim Hartnell is the most widely-published computer author in the world. Founder of the National ZX Users' Club, and founding editor of *ZX Computing* magazine, Tim has been involved over the years in a wide variety of computer activities. His published works include *The Personal Computer Guide* (Virgin Books) and *The Giant Book of Computer Games* (Fontana).

### **CLIVE GIFFORD — SERIES EDITOR AND AUTHOR**

Clive Gifford is an 18-year-old student taking his 'A' levels in Politics, Business Studies and Law. He has contributed to *ZX Computing* magazine, and has previously written *Making the Most of Your Dragon 32*. Any leisure time is spent playing golf and hockey and listening to music.

### **SUE WALLIKER — ILLUSTRATOR**

Sue Walliker is a freelance illustrator.

### **ACKNOWLEDGEMENTS**

The author wishes to thank the following for their contributions to this book: David Edwards, Phil Broughton, Adam Waring. The author also wishes to thank the following for their continual guidance and help: Peter, Sally, Liz, Tim.

**TO CATHERINE, A SOURCE OF INSPIRATION**

# CONTENTS

Editor's Introduction.....	6
Author's Introduction.....	7
Crash!.....	9
Pearl Diver.....	22
The Ring of Power.....	30
The Seven Keys of Tarkus.....	60
School's Out.....	71
Everyday Adventure.....	81
Writing Your Own Adventures.....	108
Glossary.....	114
Bibliography.....	126

# Editor's Introduction

Here is your chance to really test yourself, to trade your 'zapping alien' skills for logic, deduction and foresight and be plucked from your everyday lifestyle to survive in a situation vastly different from normal.

This book is packed with adventure games that will test your brain power and your nerve. The contributors have been selected from the very best adventure designers and programmers, and the result is a varied but always interesting collection of adventures.

The programming styles are varied too; from studying the listings you can learn much about the adventure writing process. At the back of the book, a chapter on writing your own adventures should aid you considerably.

Finally, I hope you enjoy playing and solving these adventures as much as we enjoyed writing them.

Clive Gifford, series editor  
Ashford, Middlesex  
June 1984



# Author's Introduction

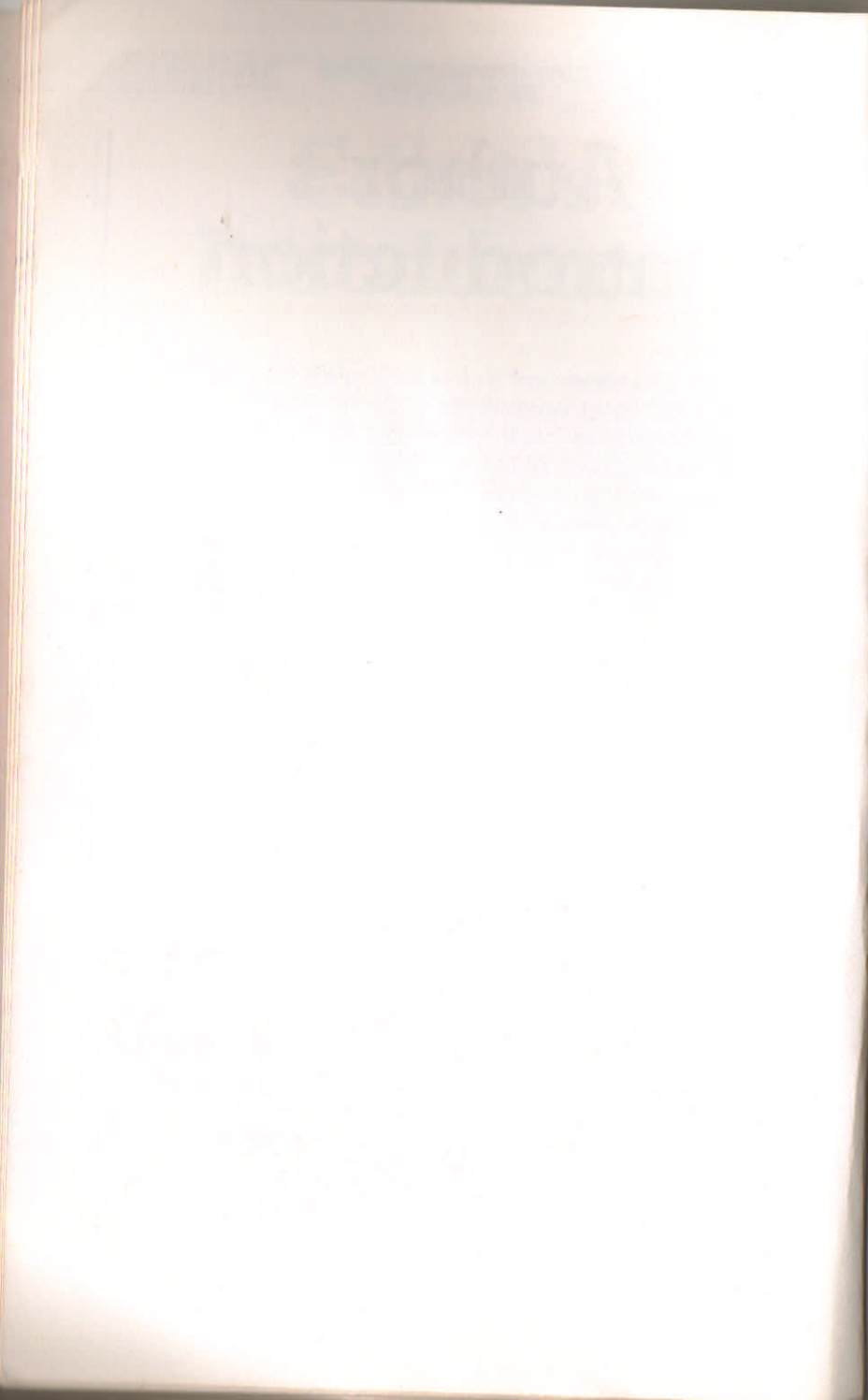
Adventure games, for me, are a superior source of computer fun. Good adventures can tax your powers of reasoning and logic to the limits and thus provide a satisfying challenge which is also entertaining.

Poor adventures, however, can be very dull and tedious; in this book, the other contributors and I have tried to offer games that cover the whole adventure scene — from graphical adventures such as 'Tarkus', to the classic 'blood and thunder' style of 'The Ring of Power'. I have tried to mix the classic type of scenario, such as the underwater adventure and the castle adventure, with novel situations such as exploring the insides of a computer, trying to escape detention at school and travelling around an everyday town.

In writing these adventures, different programming styles, concepts and scenarios were used, but one criterion was always at the forefront of my mind, are these adventures *fun* and do they provide a *challenge*. I believe that these adventures provide both and hope that you feel the same after you have typed them in and played them.

There is a technical point concerning the program listings. These have been produced on a high-quality printer, much better than the old ZX printer. However, one character appears differently on the listing than it does on the Spectrum keyboard. The hash sign on the '3' key on the Spectrum appears on the listing as a small triangle.

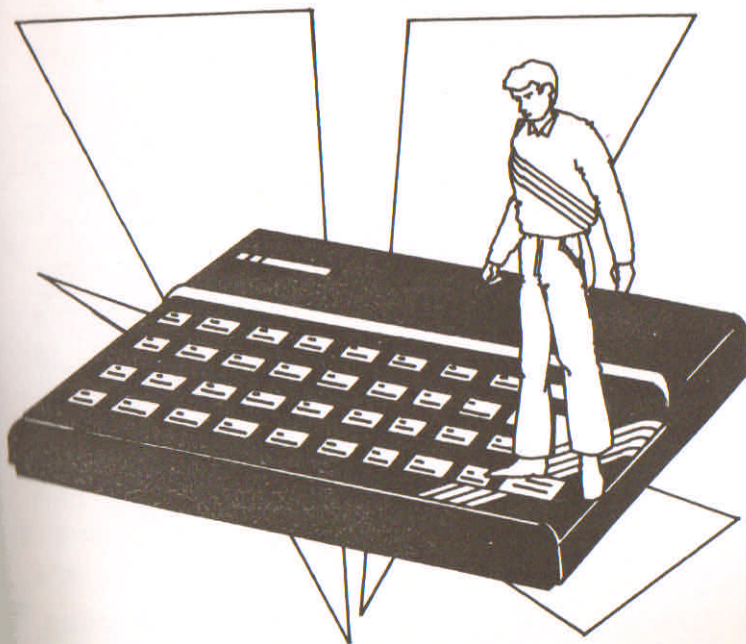
Clive Gifford  
Ashford, Middlesex  
June 1984



# CRASH!

You're playing your latest arcade game, you're just about to reach a new high score when...CRASH! Your machine has crashed, the computer has broken down and you must fix it. Your only real chance is to use the shrinking beam to become small enough to enter the computer and sort out the problem. Your technical knowledge and powers of logic will be tested to the full in this small but absorbing adventure.

The game shows how, by using a totally different scenario, you can create an adventure of much originality and excitement. Try thinking up some new scenarios for yourself or have a look at the chapter at the back of this book on writing your own adventures.



```

10 BORDER 0: INK 7: PAPER 0
20 PRINT AT 2,10: FLASH 1:" CRASH
"
30 FOR n=0 TO 100
40 POKE 22527+RND*704,RND*127
50 BEEP .005,RND*50
60 NEXT n
90 PRINT AT 10,0:" Your Computer H
as Crashed You Must Fix It !"
90 PRINT AT 17,10:" by "
100 PRINT AT 18,8:" Dave Edwards "
110 DIM k$(29,64): DIM a(29,6): DIM
o$(14,2,32): DIM o(14,2): DIM i$(5,
2,32): D
IM i(5): DIM w$(19,3)
130 LET b=0: LET z=1: LET ta=0: LET
q=0: LET c=0: LET in=1: LET s=0: LE
T li=0:
LET rm=1: LET e$="": LET f$="": LET
a=0
150 REM read data
170 FOR n=1 TO 29: READ k$(n): NEXT
n
190 FOR n=1 TO 29: READ a(n,1),a(n,
2),a(n,3),a(n,4),a(n,5),a(n,6): NEXT
n
210 FOR n=1 TO 18: READ w$(n): NEXT
n
230 FOR n=1 TO 14: READ o$(n,1),o$(
n,2),o(n,1),o(n,2): NEXT n
250 LET rm=1
270 CLS
290 REM entry
310 IF s=0 THEN LET e$=" Your comp
uter is on."

```

```

330 IF li=0 AND rm>5 THEN PRINT "
It is dark ": LET e$="": LET f$=""
350 IF li=1 OR li=0 AND rm<5 THEN
PRINT " You Are";k$(rm)
370 PRINT e$: PRINT f$
390 LET e$="": LET f$=""
410 PRINT " You see ";
430 FOR n=1 TO 14: IF o(n,1)=rm THE
N IF a=1 THEN PRINT " " +o$(
(n,1)
431 IF o(n,1)=rm AND a=0 THEN PRIN
T o$(n,1): LET a=1
432 NEXT N
450 IF a=0 THEN PRINT "nothing imp
ortant."
470 LET a=0: PRINT : PRINT
490 REM entry point 2
510 LET a=0
530 INPUT " What next ";z$
550 REM dissection
570 LET x$=" "
575 IF LEN z$<=2 THEN LET z$=z$+"
"
580 FOR x=1 TO LEN z$-LEN x$+1: IF
x$=z$(x TO x+LEN x$-1) THEN GO TO 5
90
585 NEXT x: LET x=0
590 LET a$=z$( TO x): LET b$=z$(x+1
TO )
595 LET xa=3: LET xb=3: IF LEN a$<=
3 THEN LET xa=LEN a$
600 IF LEN b$<=3 THEN LET xb=LEN b
$
605 LET c$=a$( TO xa): LET d$=b$( T
O xb)

```

```

610 FOR n=1 TO 19
630 IF c$=w$(n) THEN GO TO 710
650 IF d$=w$(n) THEN GO TO 710
670 NEXT n
690 GO TO 730
710 IF n=1 THEN GO SUB 810: LET n=
0
711 IF n=2 THEN GO SUB 830: LET n=
0
712 IF n=3 THEN GO SUB 850: LET n=
0
713 IF n=4 THEN GO SUB 870: LET n=
0
714 IF n=5 THEN GO SUB 890: LET n=
0
715 IF n=6 THEN GO SUB 910: LET n=
0
716 IF n=7 THEN GO SUB 930: LET n=
0
717 IF n=8 THEN GO SUB 930: LET n=
0
718 IF n=9 THEN GO SUB 990: LET n=
0
719 IF n=10 THEN GO SUB 1030: LET
n=0
720 IF n=11 THEN GO SUB 1070: LET
n=0
721 IF n=12 THEN GO SUB 1090: LET
n=0
722 IF n=13 THEN GO SUB 1130: LET
n=0
723 IF n=14 THEN GO SUB 1230: LET
n=0
724 IF n=15 THEN GO SUB 1250: LET
n=0

```

```
725 IF n=16 THEN GO SUB 1350: LET  
n=0  
726 IF n=17 THEN GO SUB 1450: LET  
n=0  
727 IF n=18 THEN GO SUB 1530: LET  
N=0  
728 LET C=1  
730 IF c=0 AND rm>4 THEN PRINT " Y  
ou are suddenly struck down by a pas  
sing ele  
ctron !": GO TO 3090  
750 IF c<>0 THEN LET c=0: GO TO 77  
0  
755 IF c=0 THEN GO SUB 3110: IF c=  
1 THEN GO TO 770  
760 IF c$="" THEN PRINT " Kindly r  
ephrase that !"  
765 IF c$<>"" THEN PRINT " I don't  
know how to ";b$: PRINT " dummy !"  
766 IF c$="" AND d$="" THEN PRINT  
" Kindly Rephrase That!"  
770 PAUSE 3000  
790 IF q=0 THEN GO TO 490  
795 IF q=1 THEN LET q=1: GO TO 270  
800 STOP  
810 IF a(rm,1)<>0 THEN LET rm=a(rm  
,1): PRINT " O.K. ": LET q=1: RETURN  
  
815 LET rm=rm: PRINT " O.K. ": LET  
q=1: RETURN  
830 IF a(rm,2)<>0 THEN LET rm=a(rm  
,2): PRINT " O.K. ": LET q=1: RETURN  
  
835 LET rm=rm: PRINT " O.K. ": LET  
q=1: RETURN
```

```

850 IF a(rm,3)<>0 THEN LET rm=a(rm
,3): PRINT " O.K. ": LET q=1: RETURN

855 LET rm=rm: PRINT " O.K. ": LET
q=1: RETURN

870 IF a(rm,4)<>0 THEN LET rm=a(rm
,4): PRINT " O.K. ": LET q=1: RETURN

875 LET rm=rm: PRINT " O.K. ": LET
q=1: RETURN

890 IF a(rm,5)<>0 THEN LET rm=a(rm
,5): PRINT " O.K. ": LET q=1: RETURN

895 LET rm=rm: PRINT " O.K. ": LET
q=1: RETURN

910 IF a(rm,6)<>0 THEN LET rm=a(rm
,6): PRINT " O.K. ": LET q=1: RETURN

915 LET rm=rm: PRINT " O.K. ": LET
q=1: RETURN

930 IF in>5 THEN PRINT FLASH 1;"
You can't carry any more !": RETURN

950 FOR n=1 TO 14: IF o$(n,2, TO 3)
=d$ AND rm=o(n,1) THEN LET i$(in,1)
=o$(n,1)
: LET i$(in,2)=o$(n,2): LET i(in)=o(
n,2): GO TO 970

955 IF o$(n,2, TO 3)=d$ AND rm<>o(n
,1) THEN INK 4: PRINT FLASH 1;" Yo
u don't
see that here !": INK 7: RETURN

960 NEXT n: INK 6: GO SUB 962: IF z
<>1 THEN PRINT FLASH 1;" What is a
";b$;"
?": RETURN

```



```

961 RETURN
962 IF d$="wir" OR d$="res" OR d$="
rom" OR d$="ram" THEN PRINT " You C
an't Get
That !": LET z=1: RETURN
970 PRINT " O.K. ": LET o(n,1)=-2:
LET in=in+1: IF d$="pen" THEN LET
li=1: L
ET sw=1
975 RETURN
990 FOR n=1 TO in: IF d$=i$(n,2, TO
3) THEN LET o(i(n),1)=rm: PRINT "
O.K.": G
O TO 1010
995 NEXT n: PRINT " You don't have
a ";b$;" !": RETURN
1010 FOR n=n TO in-2: LET i$(n,1)=i$
(n+1,1): LET i$(n,2)=i$(n+1,2): LET
i(n)=i(n
+1): NEXT n: LET i$(in-1,1)="": LET
i$(in-1,2)="": LET i(in-1)=0: LET in
=in-1: R
ETURN
1030 IF d$="res" AND rm=29 OR d$="wi
r" AND rm=9 THEN GO TO 1050
1035 FOR n=1 TO 14: IF d$=o$(n,2, T
O 3) THEN PRINT " It's just ";o$(n,
1): RETU
RN
1040 NEXT n: PRINT " You don't see a
ny thing like that to examine !":
RETURN
1050 IF d$="res" THEN PRINT " It's
come loose at one end !": RETURN
1055 IF d$="wir" THEN PRINT " It's

```

```

broken at one end !": RETURN
1070 LET q=1: PRINT " O.K. ": RETURN

1090 FOR n=1 TO in: IF i$(n,2, TO 3)
="clo" THEN GO TO 1110
1095 NEXT n: PRINT " What do i clean
with dummy ?": RETURN
1110 IF rm=27 AND d$="cir" THEN PR
INT " You clean the circuit. A red
glow e
mmits from it.": LET ta=ta+1: RETURN

1115 PRINT " You clean the ";b$;" ..
.": PRINT " Nothing happens.": RETUR
N
1130 IF rm<>1 THEN PRINT FLASH 1;"
Type on what ? ": RETURN
1150 IF d$<>"run" AND s<>0 THEN PRI
NT " You type ";b$;" ..": PRINT " Th
e comput
er displays 'COMMAND SYNTAX ERROR.'
": RETURN
1170 IF d$="run" AND s=0 AND ta<>5 T
HEN PRINT " The computer displays '
";: PRIN
T FLASH 1;" ROM/RAM ERROR ";: PRINT
FLASH 0;""": RETURN
1190 IF d$="run" AND s=0 AND ta=5 TH
EN PRINT " The computer is now O.K.
You
can now play your game of JET
NET BILLY!!": STOP
1210 PRINT " You type ";b$;"..": PRI
NT " The computer displays ' SYNTAX
ERROR I

```

```
N COMMAND!": RETURN
1230 PRINT " Time passes ...": RETURN
N
1250 IF rm=29 AND o(1,1)=rm OR rm=9
AND o(2,1)=rm THEN GO TO 1270
1255 PRINT " Solder what ?": RETURN
1270 FOR n=1 TO in-1: IF i$(n,2, TO
3)="sol" THEN LET a=1
1275 IF i$(n,2, TO 3)="iro" THEN L
ET b=1:
1280 NEXT n
1285 IF a=0 OR b=0 THEN PRINT " You
can't solder yet !": RETURN
1290 IF a=1 AND b=1 AND d$="res" THE
N PRINT " You solder the resistor":
LET ta=
ta+1: RETURN
1295 IF a=1 AND b=1 AND d$="wir" THE
N PRINT " You solder the wire.": LE
T ta=ta+
1: RETURN
1310 PRINT " You can't solder yet!":
RETURN
1350 CLS : PRINT AT 0,10;" Inventory
"
1370 PRINT : PRINT
1390 FOR n=1 TO in-1: IF i$(n,2, TO
3)<>" THEN LET a=1: PRINT " ";i$(
n,1)
1395 NEXT n: IF a=0 THEN PRINT " Yo
u have nothing with you !": RETURN
1410 PRINT : PRINT
1430 RETURN
1450 IF d$="rom" AND rm=25 OR d$="ra
m" AND rm=22 THEN GO TO 1490
```

```

1470 PRINT " Debug what ?": RETURN
1490 FOR n=1 TO in-1: IF i$(n,2, TO
3)<>"car" THEN NEXT n: PRINT " You
can't de
bug that !": RETURN
1495 PRINT " You debug the ";d$;" b
ug": LET ta=ta+1: IF rm=25 THEN LET
o(12,1)
=-2: RETURN
1500 IF rm=22 THEN LET o(13,1)=-1:
RETURN
1530 IF rm=1 AND d$="on" THEN LET s
=0: PRINT " O.k.": RETURN
1535 IF d$="off" THEN LET s=1: PRIN
T " O.K.": RETURN
1540 PRINT FLASH 1;" Switch what ?"
: RETURN
1550 REM DATA ( locations )
1570 DATA " Outside The Computer"
1590 DATA " In The Shrink Beam"
1610 DATA " On A Centronics Input/
Output Port"
1630 DATA " On The Data Bus"
1650 DATA " At A Logic Gate"
1670 DATA " At A Logic Gate"
1690 DATA " On The Data Bus"
1710 DATA " In The Power On/Off Unit"
1730 DATA " At The Keyboard
Interface Unit"
1750 DATA " At The Video Modulator"
1770 DATA " On The Data Bus"
1790 DATA " Below The Main
Processor"
1810 DATA " Surrounded By The Main
Processor"

```

1830 DATA " Surrounded By The Main  
 Processor"  
 1850 DATA " Surrounded By The Main  
 Processor"  
 1870 DATA " Surrounded By The Main  
 Processor"  
 1890 DATA " Between Two Memory  
 Chips"  
 1910 DATA " On The Data Bus"  
 1930 DATA " On The Data Bus"  
 1950 DATA " By The Rampack  
 Connector"  
 1970 DATA " On The Rampack  
 Connector"  
 1990 DATA " In The Rampack"  
 2010 DATA " In The Rampack"  
 2030 DATA " In The Rampack"  
 2050 DATA " In The Rampack"  
 2070 DATA " In The Rampack"  
 2090 DATA " Standing On A Dirty  
 Circuit Connector"  
 2110 DATA " Above A Cartridge Port"  
 2130 DATA " In A Mass Of Resistors"  
 2150 REM DATA ( locations exits )  
 2170 DATA 2,0,0,0,0,0  
 2190 DATA 3,1,1,1,0,0  
 2210 DATA 4,2,1,1,0,0  
 2230 DATA 7,3,6,5,0,0  
 2250 DATA 8,9,4,10,0,0  
 2270 DATA 28,0,29,4,0,0  
 2290 DATA 11,4,6,8,0,0  
 2310 DATA 0,5,7,27,0,0  
 2330 DATA 5,0,0,0,0,0  
 2350 DATA 27,0,5,0,0,0  
 2370 DATA 12,7,0,0,0,0

```

2390 DATA 0,11,0,0,14,0
2410 DATA 0,14,15,0,0,0
2430 DATA 13,0,16,0,0,12
2450 DATA 0,16,0,13,0,0
2470 DATA 15,0,17,14,0,0
2490 DATA 18,0,0,16,0,0
2510 DATA 0,17,19,0,0,0
2530 DATA 0,0,20,18,0,0
2550 DATA 21,0,0,19,0,0
2570 DATA 22,20,0,0,0,0
2590 DATA 0,21,0,23,0,0
2610 DATA 26,24,22,25,0,0
2630 DATA 23,0,0,0,0,0
2650 DATA 0,0,23,0,0,0
2670 DATA 0,23,0,0,0,4
2690 DATA 0,10,0,0,0,0
2710 DATA 0,6,0,0,0,0
2730 DATA 0,0,0,6,0,0
2750 REM DATA (COMMANDS)
2770 DATA "nor","sou","eas","wes","u
p","dow","get","tak","dro","exa","lo
o","cle"
,"typ","wai","sol","lis","deb","swi"
2790 REM DATA (objects)
2810 DATA "A Resistor","",29,1
2820 DATA "A Wire","",9,2
2830 DATA "A Soldering Iron","iro",1
1,3
2840 DATA "A Light Pen","pen",4,4
2850 DATA "A Debugging Cartridge","c
ar",5,5
2860 DATA "Some Tin Solder","sol",19
,6
2870 DATA "A Cloth","clo",0,7

```

CRASH!

```
2880 DATA "A Rom Chip","rom",12,8
2890 DATA "A Timing Crystal","cry",2
6,9
2900 DATA "A Capicitor","cap",20,10
2910 DATA "An Old Triode","tri",29,1
1
2920 DATA "A Rom Bug","",25,12
2930 DATA "A Ram Bug","",22,13
2940 DATA "A '6821' Input/Output
Chip","chi",10,14
3090 INPUT " Do You Want To Play Aga
in ?";a$: IF a$<>"y" AND a$<>"yes" T
HEN STO
P
3095 RUN
3110 .RETURN
```

# PEARL DIVER

In this adventure, you assume the role of a humble fisherman who, after hearing some talk about pearls in the south cove, has rushed out his boat and is now looking for treasure. You need to collect five oysters to become rich but the sea is full of dangers, from piranhas to seaweed.

You must collect each oyster separately and swim back up to your boat; the oysters are a phenomenal size and can only be carried one at a time. Your oxygen strength at the beginning of the adventure is 90, and this goes down as you swim around and get caught in hazards. However, you may be lucky enough to pick up an unused air tank that is lying on the seabed.

The game features a grid system of movement. This sort of game is normally much easier to solve by mapping out the adventure on some paper and just moving round the grid. I have made this adventure more difficult by adding a time limit (90 moves) and by not having a uniform grid. Instead of a straightforward cube my grid is laid out a little differently. It can still be mapped and I suggest you do this since it is the best way to solve the adventure, but it will take more time and concentration.

Good luck and keep on searching for those giant oysters!





## PEARL DIVER

```
10 REM *****PEARL DIVER*****
20 GO SUB 5000
30 LET L=3: LET S=90
40 LET TRE=0
90 PAPER 1: INK 7: BORDER 5
100 REM *****GAME ACTION*****
110 CLS
120 PRINT : PRINT "
IVER          "
124 PRINT : PRINT : PRINT : PRINT "
YOU ARE ";
130 IF L<17 THEN PRINT "IN THE SHA
LLOWS"
132 IF L<33 AND L>16 THEN PRINT "
SEVERAL FATHOMS DOWN"
134 IF L<49 AND L>32 THEN PRINT "I
N THE MURKY DEPTHS"
136 IF L>48 THEN PRINT "ON THE OC
EAN FLOOR"
140 PRINT : PRINT "YOUR STRENGTH IS
";S
150 PRINT : PRINT "YOU CAN SEE ";
170 GO TO 1000+A(L)*50
180 PRINT : INPUT "WHICH DIRECTION?
(N,S,E,W)";M$
190 IF M$<>"N" AND M$<>"S" AND M$<>
"E" AND M$<>"W" AND M$<>"D" AND M$<>
"U" THEN
PRINT "NO SUCH DIRECTION": LET S=S
-1: GO TO 180
200 IF M$="N" AND L>8 THEN LET L=L
-S
210 IF M$="S" AND L<56 THEN LET L=
L+8
```

ADVENTURES FOR YOUR  
ZX SPECTRUM

```

220 IF M$="E" AND L>1 THEN LET L=L
-1
230 IF M$="W" AND L<64 THEN LET L=
L+1
240 LET S=S-1: IF S<=0 THEN GO TO
2300
300 GO TO 90
340 PRINT AT 8,15;" "
1000 REM *****CLEAR*****
1010 PRINT "NOTHING"
1020 GO TO 180
1050 REM *****OCTOPUS*****
1060 PRINT FLASH 1;"AN OCTOPUS!!!"
1070 LET K=18
1080 GO SUB 2000
1090 GO TO 180
1100 REM *****SHARK*****
1110 PRINT FLASH 0;"A MANEATING SHA
RK"
1120 LET K=32
1130 GO SUB 2000
1140 GO TO 180
1150 REM *****GIANT SQUID*****
1160 PRINT "GULP! A GIANT SQUID"
1170 LET K=24
1180 GO SUB 2000
1190 GO TO 180
1200 REM *****SEAWEEED*****
1210 PAPER 4: PRINT "S E A W E E D"
1215 PAPER 1
1220 PRINT "... YOU MANAGE TO UNTANG
LE YOURSELF BUT YOU LOSE 5 STRENGTH
POINTS
IN DOING SO.": LET S=S-5

```

## PEARL DIVER

```
1230 PAUSE 150
1240 GO TO 180
1250 REM *****OYSTERS*****
1260 PRINT FLASH 1;"A N O Y S T
E R"
1270 LET TRE=TRE+1: LET A(L)=0
1280 BEEP .5,20: BEEP .5,35
1290 GO TO 180
1300 REM *****OLD CHEST*****
1310 PRINT "AN OLD CHEST": PRINT "DO
YOU WISH TO OPEN IT (Y/N)?"
1320 INPUT V$: IF V$<>"N" AND RND>.6
& THEN PRINT ,,"A HIDEOUS SPIRIT LE
APS OUT
AND KILLS YOU...": PAUSE 150: GO TO
2300
1330 IF V$<>"N" THEN PRINT ,,"YOU H
AVE FOUND MUCH TREASURE... GO BACK T
O YOUR B
DAT A RICH MAN": LET TRE=5
1340 GO TO 180
1350 REM *****PIRANHAS*****
1360 PRINT FLASH 1;"PIRANHAS!!!"
1370 LET K=14
1380 GO SUB 2000
1390 GO TO 180
1400 REM *****AIR TANK*****
1410 PRINT "AN OXYGEN CYLINDER"
1420 PRINT ,,"IT ADDS TO YOUR STRENG
TH": LET S=S+15: LET A(L)=0
1430 BEEP 1,30
1440 GO TO 180
1450 REM *****FISH*****
1460 PRINT "A SHOAL OF FISH..."
```

```

1470 PAUSE 100
1480 PRINT "DON'T WORRY, THEY ARE HARMLSS"
1490 GO TO 180
1500 REM *****BOAT*****
1510 PRINT "YOUR BOAT."
1520 PRINT "YOU DUMP YOUR TREASURES
      IN THE BOAT."
1530 IF TRE>=5 THEN GO TO 2300
1540 PRINT "YOU NOW NEED ";5-TRE;" OYSTERS"
1550 GO TO 180
2000 REM *****CONFLICT*****
2005 LET C=INT (RND*10)+1
2010 PRINT : PRINT "SWIM AWAY (S) OR
      FIGHT (F)?"
2012 INPUT A$: IF A$<>"S" AND A$<>"F"
      THEN PRINT "I CANNOT DO THAT FOOL
      !!!": GO
      TO 2012
2015 IF A$="S" AND RND>.5 THEN GO TO
      3000
2017 IF A$="F" THEN GO TO 2050
2020 IF A$="S" THEN PRINT ", "YOU GO
      T AWAY LOSING ";C;" STRENGTH": LET S
      =S-C: IF
      A(L-1)=0 AND L>1 THEN LET L=L-1: R
      ETURN
2022 IF A(L+1)=0 AND L<64 THEN LET
      L=L+1: RETURN
2024 IF A(L+8)=0 AND L<56 THEN LET
      L=L+8: RETURN
2026 IF A(L-8)=0 AND L>8 THEN LET L
      =L-8: RETURN

```

## PEARL DIVER

```

2028 LET A(L)=0: RETURN
2050 PAPER 2: CLS
2060 PRINT "      LET THE BATTLE BEGIN
...."
2070 REM *****BATTLE LOOP*****
2090 IF RND>.66 THEN PRINT "THE CRE
ATURE HURTS YOU...UGH!!!": LET S=S-1
2100 IF RND<.34 THEN PRINT "YOU LAU
NCH A WELL-TIMED ATTACK": LET K=K-(I
NT (RND*
4)+1)
2110 PRINT ,,,,"CREATURE:";K;"      Y
OU:";S
2120 IF S<=0 THEN GO TO 3000
2130 IF K<=0 THEN FOR T=1 TO 10: B
ORDER 3: PAUSE 30: BORDER 5: PAUSE 3
0: NEXT
T: PRINT ,,,,"      YOU BEAT HIM
": LET A(L)=0: R
ETURN
2140 PRINT : PRINT : GO TO 2090
2300 REM *****VICTORY*****
2320 FOR T=0 TO 7: BORDER T: PAPER T
: BEEP .6,T*5: NEXT T
2330 INK 1: PRINT AT 10,10: FLASH 1;
BRIGHT 1;"You're Rich!"
2350 BEEP 3,40
2360 FOR T=1 TO 200: PRINT AT INT (R
ND*8)+12,INT (RND*30)+1: INK RND*7;"
E"
2365 PRINT AT INT (RND*8)+1,INT (RND
*30)+1: INK RND*7;"E"
2370 BEEP .009,RND*55: NEXT T
2900 STOP

```

```

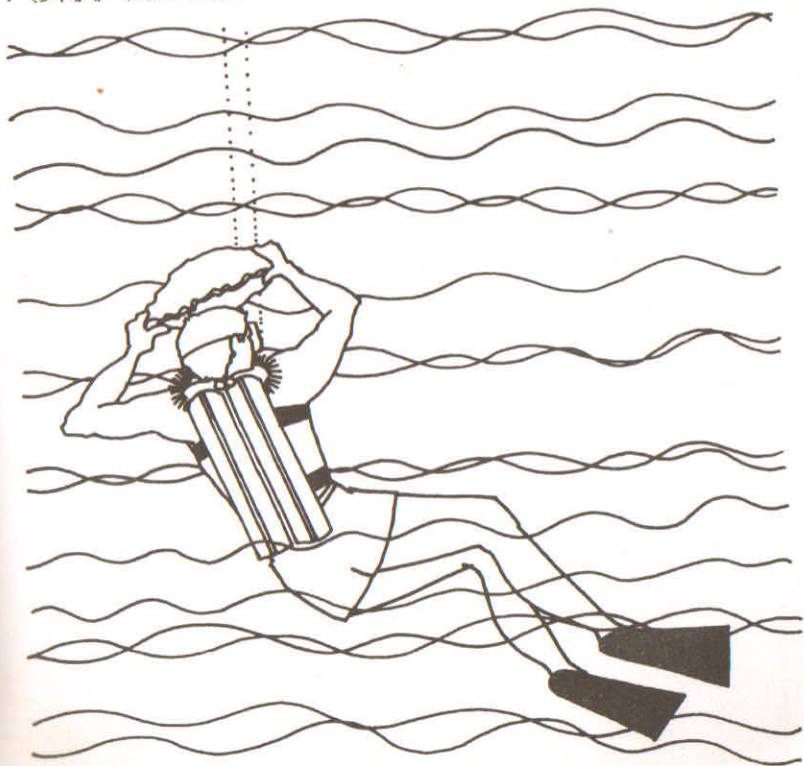
3000 REM *****DEFEAT*****
3005 PAPER 0: INK 7: BORDER 1: CLS
3030 OVER 0: INVERSE 1
3040 PRINT AT 8,15;" "
3050 PRINT AT 9,15;" "
3060 PRINT AT 10,12;" R.I.P "
3070 PRINT AT 11,15;" "
3080 PRINT AT 12,15;" "
3090 PRINT AT 13,15;" "
3092 PRINT AT 14,15;" "
3094 PRINT AT 15,15;" "
3100 PRINT AT 16,13;"      "
3110 PRINT AT 17,11;"      "
3120 INVERSE 0: OVER 1
3150 PRINT AT 2,2;"You fall to a wat
ery grave"
3160 PRINT "  Your greed for pearls
has"
3170 PRINT "  left you in Davy Jones
Locker"
3190 BEEP 1,0: BEEP 1,4: BEEP 2,0
3900 STOP
5000 REM *****DATA*****
5010 RESTORE
5020 DIM A(65)
5030 FOR X=2 TO 65: READ A(X): NEXT X
5050 DATA 1,0,0,4,0,2,0,9,0,4,0,0,4,
0,2,10
5060 DATA 7,0,0,4,0,0,7,0,3,9,0,0,0,
7,0,7
5070 DATA 0,3,0,5,1,0,2,0,8,0,0,9,2,
0,3,0
5080 DATA 5,0,8,5,0,4,5,0,0,4,6,0,5,
0,0,5

```

PEARL DIVER

```

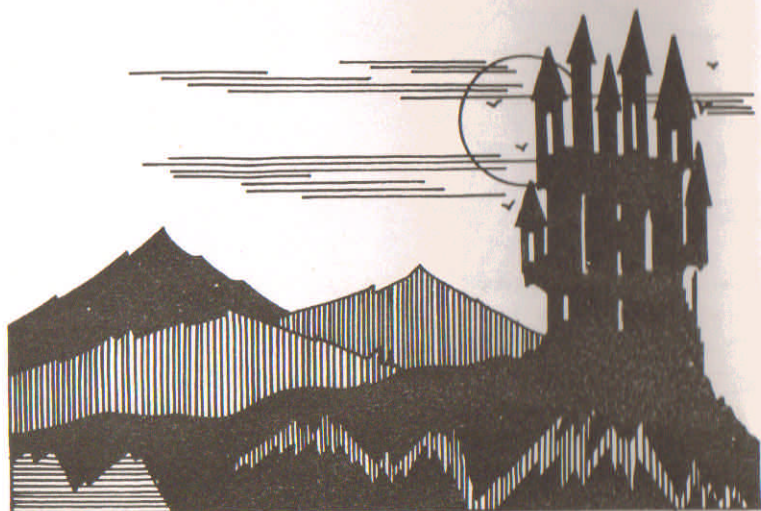
5000 REM *****TITLE*****
5010 INK 7: PAPER 1: CLS
5020 PRINT AT 8,5;"P E A R L   D I
V E R"
5040 FOR T=1 TO 20: CIRCLE 125,30,T:
NEXT T
5050 BEEP 2,20: BEEP 2,30: BEEP 2,25
5200 PAUSE 5000
5210 BEEP 1,10
5300 REM #####
#####
#####
5400 RETURN
  
```



# THE RING OF POWER

This massive adventure was the result of much intensive work by London programmer, David Edwards. You are placed in a mythical world where cunning and courage are essential to reach your goal. You are looking for the great Ring of Power, for whoever holds this holds the key to controlling all of the underworld. You have reached the foreboding castle where the ring is reputed to lie. You must collect the various Source Keys, for without these your journey through the castle will be obstructed.

This adventure is extremely difficult. There are less puzzles to solve than in the Everyday adventure, but movement around the adventure has to be made carefully. There are several large mazes which are very hard to escape from.





When entering commands, limit them to a couple of words. The computer may not recognise your command but it may recognise a slightly different format. For example, the computer may not accept GET THE PLATINUM KEY but it will accept GET PLATINUM as there are a number of keys to find. Use your judgement on these matters. It is worth noting down every move you make so that you can refer to them later. Believe me, this adventure will not just take a couple of hours to solve, it is a real challenge not far behind the offerings of the commercial software companies.

On account of its large size, the adventure has to be loaded in two parts. The procedure is as follows:

- 1) Type in program 1.
- 2) Save program 1 on tape using the format, SAVE "RING" LINE 1.
- 3) Type in program 2.
- 4) Save program 2 on tape immediately after program 1.
- 5) RUN program 2.

Once this is all on tape, the next time you wish to use the program, just type LOAD"" and the whole adventure will load itself automatically.

If you are asked at the end of the game whether you wish to play again and you do then type 'Y' and reload the second program.

One final note: since this adventure is so tough, I shall give you one clue. For the first maze, 'NWNNEESS' is of use.

```

5 REM *****THE RING OF POWER*****
***
10 DIM I$(9,2,32): DIM I(9,2): DIM
D(5)
14 LET F$="": LET QW=0
15 DIM D$(8,64): DIM K(6)
16 BORDER 4: PAPER 1: INK 7
    
```

```

17 LOAD "" DATA P$(): LOAD "" DATA
P(): LOAD "" DATA O$(): LOAD "" DAT
A O(): L
OAD "" DATA W$()
190 PAPER 1: INK 7: BORDER 4
200 CLS
210 FOR N=25 TO 155 STEP 5
220 CIRCLE N,N,5
225 CIRCLE 255-N,N,5
230 NEXT N
250 PRINT AT 21,6: FLASH 1:" THE RI
NG OF POWER "
260 FOR N=0 TO 3000: NEXT N
530 LET RM=1: LET SA=0: LET SB=0: L
ET IN=1: LET D=0: LET Q=1000: LET E$
="": LET
GR=1: LET A$="": LET W=0: LET Z=555
: LET D=0: LET R=0: LET AL=0: LET CE
=0: LET
EW=0
540 LET D$(1)=" YOU DON'T SEE THAT
HERE!": LET D$(2)=" YOU CAN'T GO THA
T WAY!":
LET D$(3)=" WHAT IS A ": LET D$(4)=
" YOU CAN'T TAKE THAT!": LET D$(5)="
YOU DON
'T HAVE THAT!": LET D$(6)=" DON'T BE
SILLY": LET D$(7)=" O.K"
550 IF RM=6 AND SR=1 THEN LET A$
=" AS YOU ENTER A PORTCULLIS C
LOSES IN
ON YOU!": LET SR=SR+1
555 IF RM=16 AND SR=2 THEN LET A
$=" AS YOU ENTER A VOICE SAYS 'WELL

```

DONE PUN

Y ONE!"; LET SR=SR+1

560 IF RM=40 AND D=0 THEN LET E\$="  
A LARGE VISCIOUS DOG STANDS IN THE  
CORNER.

"

570 IF RM=43 AND R=0 THEN LET E\$="  
" A LARGE RABBIT STANDS IN THE WA  
Y OF THE  
DOOR"

580 IF RM=17 AND D(1)=0 THEN LET  
E\$=" THE EAST EXIT IS BLOCKED BY A  
LARGE D  
OOR"

590 IF RM=29 AND D(2)=0 THEN LET E  
\$=" THE NORTH EXIT IS BLOCKED BY A  
LARGE DO  
OR"

500 IF RM=37 AND D(3)=0 THEN LET  
E\$=" THE EAST EXIT IS BLOCKED BY A  
SOLID S  
TEEL DOOR"

510 IF RM=40 AND D(4)=0 THEN LET  
E\$=" THE NORTH EXIT IS BLOCKED BY A  
SOLID S  
TEEL DOOR"

520 IF RM=69 AND D(5)=0 THEN REM \$  
=" THE ENTRANCE TO THE RING OF P  
OWER IS  
BLOCKED BY A MASSIVE WOODEN DOOR"

530 IF RM=21 AND AL=0 THEN LET E\$  
=" YOU SEE THE BODY OF A YOUNG G  
IRL ON T  
HE ALTAR, SLAIN."

540 IF RM=21 AND AL=1 THEN LET E\$

```

=" YOU SEE THE BODY OF A YOUNG      G
IRL ON T
HE FLOOR."
  550 CLS
  560 FOR N=1 TO 40: IF O(N,1)=RM THE
N LET F#=F#+O*(N,1)
  561 NEXT N
  570 IF CE=0 AND RM>44 AND RM<60
THEN PRINT "          IT'S DARK.":
PRINT :
LET F#="": LET E#="": LET A#="": GO
TO 690
  580 IF CE<>0 THEN GO TO 702
  590 IF RM>44 AND RM<58 THEN BEEP @
.1,1: LET EW=EW+1
  591 IF RM<44 OR RM>60 THEN GO TO 7
02
  592 IF RM>58 AND RM<60 THEN BEEP @
.1,50: LET EW=EW+1
  593 IF EW=16 THEN GO TO 700
  594 IF EW<>16 AND CE<>1 THEN GO TO
800
  595 IF EW<>16 AND CE=1 THEN GO TO
702
  700 PRINT " YOU ARE SUDDENLY ATTACK
ED BY A GIANT SPIDER. YOU ARE NOW V
ERY DEA
D!": PRINT : PRINT " DO YOU WAN'T TO
PLAY AGAIN ? (Y/N)": INPUT Z#: I
F Z#<>"Y
" THEN STOP
  701 IF Z#="Y" THEN RUN
  702 FOR N=1 TO 192: IF P*(RM,N TO N
)=" " THEN LET QW=QW+1

```

```

703 IF P$(RM,N TO N)<>" " THEN LET
QW=0
704 IF QW=15 THEN GO TO 710
705 NEXT N
706 LET N=192
710 IF RM=34 THEN LET N=192
711 PRINT AT 0,0;" YOU ARE";P$(RM,
TO N)
720 PRINT A$: PRINT E$: LET A$="":
LET E$=""
730 PRINT : PRINT " YOU SEE ";
735 IF F$="" THEN PRINT "NOTHING I
MPORTANT": GO TO 750
740 PRINT F$
745 LET F$=""
750 PRINT " EXITS ARE :-";
752 IF P(RM,1)<>0 THEN PRINT " NOR
";
753 IF P(RM,2)<>0 THEN PRINT " SOU
";
754 IF P(RM,3)<>0 THEN PRINT " EAS
";
755 IF P(RM,4)<>0 THEN PRINT " WES
";
756 IF P(RM,5)<>0 THEN PRINT " UP"
;
757 IF P(RM,6)<>0 THEN PRINT " DOW
";
758 PRINT
760 IF RM=3 OR RM=26 THEN FOR N=0
TO 1000: NEXT N: PRINT " DO YOU WAN'
T TO PLA
Y AGAIN ? (Y/N).": INPUT Z$: IF Z
$="Y" THEN RUN

```

```

762 IF RM=3 OR RM=26 THEN IF Z$="N
" THEN STOP
900 INPUT " WHAT NEXT ? ";C$
905 LET X$=""
906 IF LEN (C$)<=2 THEN LET C$=C$+
"
"
910 FOR X=1 TO LEN C$-LEN X$+1: IF
X$=C$(X TO X+LEN X$-1) THEN GO TO 8
30
920 NEXT X: LET X=0: GO TO 830
930 LET T$=C$( TO X): LET Q$=C$(X+1
TO )
931 LET XA=3: LET XB=3
935 IF LEN T$<=3 THEN LET XA=LEN (
T$)
936 IF LEN Q$<=3 THEN LET XB=LEN (
Q$)
940 LET R$=T$( TO XA): LET S$=Q$( T
O XB)
945 IF LEN S$<=2 THEN LET S$=""
950 IF RM<>40 OR RM<>24 THEN GO T
O 860
952 IF R$="GET" OR R$="TAK" AND S$=
"WER" OR S$="WOL" OR S$="DOG" THEN
PRINT D$
(4): GO TO 3000
960 FOR N=1 TO 16: IF R$=W$(N) OR
S$=W$(N) AND N=11 OR S$=W$(N) AND N=
12 OR S$
=W$(N) AND N=13 OR S$=W$(N) AND N=14
OR S$=W$(N) AND N=15 OR S$=W$(N) AN
D N=16 O
R S$=W$(N) AND N=7 OR S$=W$(N) AND N
=8 THEN GO TO 870

```

```
861 NEXT N
862 GO TO 900
870 REM COMMAND SECTION
871 IF N=1 OR N=2 THEN GO SUB 4000
: GO TO 900
872 IF N=3 THEN GO SUB 4500: GO TO
900
873 IF N=4 THEN GO SUB 5000: GO TO
900
874 IF N=5 THEN GO SUB 5050: GO TO
900
875 IF N=6 THEN GO SUB 5270: GO TO
900
876 IF N=7 THEN GO SUB 5380: GO TO
3000
877 IF N=8 THEN GO SUB 5390: GO TO
3000
878 IF N=9 THEN GO SUB 5610: GO TO
900
879 IF N=10 THEN GO SUB 5700: GO T
O 900
880 IF N=11 THEN GO SUB 5490: GO T
O 3000
881 IF N=12 THEN GO SUB 5510: GO T
O 3000
882 IF N=13 THEN GO SUB 5530: GO T
O 3000
883 IF N=14 THEN GO SUB 5550: GO T
O 3000
884 IF N=15 THEN GO SUB 5570: GO T
O 3000
885 IF N=16 THEN GO SUB 5590: GO T
O 3000
886 IF RM=24 AND WK>1 THEN FOR N=0
```

```

TO 2000: NEXT N: GO SUB 7490
987 LET SA=0: LET SB=0
900 IF S$="REA" AND RM=21 THEN PRI
NT " THE ALTAR SLOWLY OPENS, THE
YOUNG G
IRL FALLS OFF THE ALTAR AND A KEY I
S REVEALED.": LET AL=1: LET O(3,1)=R
M: GO TO
3000
910 IF S$="CAR" AND RM=83 THEN PRI
NT " THE POND SLOWLY OPENS...": LET
P(RM,3)=
94: GO TO 3000
920 IF S$="LIF" AND RM=82 THEN PRI
NT " THE ALTAR CRACKS AND SPLITS
DOWN TH
E MIDDLE.": LET P(RM,2)=88: GO TO 30
00
930 IF S$="BIR" AND RM=89 THEN PR
INT " THE ALTAR SLOWLY CRACKS AND
SPLITS
DOWN THE MIDDLE..": LET P(RM,2)=93:
GO TO 3000
940 IF Q$="VAMPIRE" AND RM=96 THE
N PRINT " THE WEST WALL CRACKS AND
SPLITS D
OWN THE MIDDLE.": LET P(RM,4)=97: GO
TO 3000
950 IF S$="REA" AND RM=92 THEN PR
INT " A HOLE OPENS UP IN THE FLOOR
AND YO
U FALL DOWN IT!": GO TO 7500
960 IF Q$="5381900" AND RM=41 THEN
PRINT " A PANEL OPENS UP IN THE MAS
SIVE COM

```



```

PUTER..": LET P(RM,1)=42: GO TO 3000
  970 IF R$="DRO" AND S$="BON" AND RM
=40 THEN IF O(1,1)=RM THEN PRINT "
  THE DOG
  PICKS UP THE BONE AND WALKS AWAY
WITH IT.": LET D=1: LET O(1,1)=-2: L
ET O(2,
1)=-2: GO TO 3000
  980 IF R$="DRO" AND S$="COI" AND RM
=34 THEN IF O(4,1)=RM THEN PRINT "
  YOU HEA
R A CHINK AS THE COINS FALL DOWN T
HE WELL. YOU ARE BLINDED BY A FL
ASH! WHE
N YOU CAN SEE AGAIN TWO MORE DOOR
S HAVE BEEN REVEALED.": LET P(RM,
1)=36: L
ET P(RM,4)=35: GO TO 1000
  990 GO TO 1020
1000 LET O(4,1)=-2: GO TO 3000
1010 REM *****
1020 IF R$="DRO" AND S$="CAR" AND RM
=43 THEN IF O(25,1)=RM THEN PRINT
" THE RA
BBIT RUNS OFF WITH THE CARROTS.":
LET P(RM,4)=80: LET O(25,1)=-2: LET
R=1: GO
TO 3000
1030 IF R$="CLO" AND S$="SWI" AND RM
=80 THEN PRINT " AS YOU CLOSE THE S
WITCH A
RED GLOW EMMITS FROM THE CEILING."
: LET CE=1: GO TO 3000
1040 IF R$="TAK" AND S$="RIN" AND RM

```

```

=72 THEN GO TO 7530
1050 IF R$="GET" AND S$="RIN" AND RM
=72 THEN GO TO 7530
1060 IF S$="HEL" THEN PRINT " "; F
LASH 1: PRINT "GOD HELPS THOSE WHO H
ELP";: F
LASH 0: PRINT " ";: FLASH 1:
PRINT "THEMSELVES.": FLASH 0: GO TO
3000
1070 IF Q$="MINDSTONE" AND RM=97 THE
N FOR N=1 TO IN: IF I$(N,1, TO 3)="
MIN" THE
N LET RM=31: PRINT " THERE IS A BLI
NDING FLASH!": GO TO 3000
1071 IF Q$="MINDSTONE" AND RM=97 THE
N IF I$(N,1, TO 3)<>"MIN" THEN NEX
T N
1072 IF Q$="MINDSTONE" AND RM=97 THE
N PRINT FLASH 1;" NOTHING HAPPENS"
1080 FOR N=1 TO 16
1081 IF R$=W$(N) AND S$="" THEN PRI
NT FLASH 1;" CAN YOU COUNT ? USE TW
O WORDS"
: GO TO 3000
1082 IF S$=W$(N) AND R$="" THEN PRI
NT FLASH 1;" CAN YOU COUNT ? USE TW
O WORDS"
: GO TO 3000
1083 IF S$=W$(N) THEN GO TO 3000
1084 IF R$=W$(N) THEN GO TO 3000
1085 NEXT N
1090 IF R$="SCR" AND O(20,1)=RM THEN
PRINT " YOU CAN'T SHE'S MINE!": GO
TO 3000
1100 IF S$="FUC" OR R$="FUC" THEN I

```

```

NK 2: PRINT FLASH 1;" I'D LIKE TO!"
: INK 7:
GO TO 3000
1110 IF S$="SHI" OR R$="SHI" THEN I
NK 4: PRINT FLASH 1;" I DON'T FEEL
LIKE IT!
"; FLASH 0: INK 7: GO TO 3000
1120 IF S$="BAL" OR R$="BAL" THEN I
NK 5: PRINT FLASH 1;" I HAVE TWO OF
THEM!";
FLASH 0: INK 7: GO TO 3000
1125 IF S$="" OR R$="" THEN PRINT
FLASH 1;" CAN YOU COUNT ? USE TWO WO
RDS": GO
TO 3000
1130 INK 3: PRINT FLASH 1;" I DON'T
KNOW HOW TO ";
1135 IF T$(">") THEN PRINT FLASH 1;
T$
1136 INK 7
3000 PAUSE 30000: GO TO 550
4000 GO SUB 4100
4001 FOR N=1 TO 40
4002 IF O$(N,2, TO 3)=S$ AND O(N,1)=
RM THEN LET I$(IN,1)=O$(N,2): LET I
$(IN,2)=
O$(N,1): LET I(IN,1)=O(N,2): LET O(N
,1)=-2: PRINT D$(7): GO TO 4200
4010 IF S$=O$(N,2, TO 3) AND O(N,1)<
>RM THEN PRINT D$(1): RETURN
4020 NEXT N: PRINT D$(3, TO 8);" ";O
$;" ?"
4030 RETURN
4100 IF IN>=8 THEN GO TO 4200

```

```

4150 IF IN<=7 THEN RETURN
4200 LET IN=IN+1: IF IN>=8 THEN PRINT FLASH 1;" YOU CAN'T CARRY ANY MORE!"; FL
ASH 0: RETURN
4210 IF S$="IRO" THEN LET K(1)=1
4220 IF S$="KEY" THEN LET K(2)=1
4230 IF S$="PLA" THEN LET K(3)=1
4240 IF S$="GOL" THEN LET K(4)=1
4250 IF S$="MID" THEN LET K(5)=1
4260 RETURN
4500 FOR N=1 TO IN+1: IF S$=I$(N,1,
TO 3) THEN LET O(I(N,1),1)=RM: PRINT D$(7):
GO TO 4550
4510 NEXT N: PRINT D$(5): RETURN
4550 FOR I=N TO IN: LET I$(I,2)=I$(I
+1,2): LET I$(I,1)=I$(I+1,1): LET I(
I,1)=I(
+1,1): LET I(I,2)=I(I+1,2): NEXT I:
LET IN=IN-1: LET I$(I-1,1)="": LET I
$(I-1,2)
="": LET I(I-1,1)=-1: LET I(I-1,2)=-
1: RETURN
5001 PRINT D$(6): RETURN
5050 IF RM=40 OR RM=29 OR RM=68 OR R
M=17 OR RM=37 THEN GO TO 5150
5100 PRINT " WHERE IS THE DOOR?": RE
TURN
5150 IF RM=17 AND K(1)=1 THEN LET P
(RM,3)=27: LET D(1)=1: PRINT D$(7):
RETURN
5160 IF K(1)<>1 AND RM=17 THEN PRIN
T " YOU DON'T HAVE THE CORRECT KEY!"
: RETURN

```

5170 IF RM=29 AND K(2)=1 THEN LET P  
 (RM,1)=30: LET D(2)=1: PRINT D\$(7):  
 RETURN

5180 IF RM=29 AND K(2)<>1 THEN PRIN  
 T " YOU DON'T HAVE THE CORRECT KEY!"  
 : RETURN

5190 IF RM=37 AND K(3)=1 THEN LET P  
 (RM,3)=81: LET D(3)=1: PRINT D\$(7):  
 RETURN

5200 IF RM=37 AND K(3)<>1 THEN PRIN  
 T " YOU DON'T HAVE THE CORRECT KEY!"  
 : RETURN

5210 IF RM=40 AND K(4)=1 THEN LET P  
 (RM,1)=41: LET D(4)=1: PRINT D\$(7):  
 RETURN

5230 IF RM=40 AND K(4)<>1 THEN PRIN  
 T " YOU DON'T HAVE THE CORRECT KEY!"  
 : RETURN

5240 IF RM=68 AND K(5)=1 THEN LET P  
 (RM,1)=72: LET D(5)=1: PRINT D\$(7):  
 RETURN

5250 IF RM=68 AND K(5)<>1 THEN PRIN  
 T " YOU DON'T HAVE THE CORRECT KEY!"  
 : RETURN

5260 PRINT " YOU DON'T HAVE THE CORR  
 ECT KEY!": RETURN

5270 IF S\$<>O\$(2,2, TO 3) AND S\$<>O\$  
 (6,2, TO 3) AND S\$<>"PRI" AND S\$<>"P  
 AI" AND

S\$<>"ALT" AND S\$<>"MEN" AND S\$<>"BUL  
 " THEN INK 6: PRINT FLASH 1:" YOU  
 CAN'T EX  
 AMINE THAT": FLASH 0: INK 7: RETURN

5280 IF RM=0(2,1) OR RM=0(6,1) OR RM  
=0(20,1) OR RM=32 OR RM=38 OR RM=21  
OR RM=79

OR RM=69 THEN GO TO 5300

5290 PRINT FLASH 1;" THERE ISN'T AN  
Y THING LIKE THAT IN THE ROO  
M!

": FLASH 0

5300 FOR N=1 TO 40: IF S#=0\$(N,2, TO  
3) AND N=2 AND O(N,1)<>-1 THEN PRI  
NT " THE

BIBLE SAYS ' DEATH HOLDS THE KEY  
'": RETURN

5310 IF S#=0\$(N,2, TO 3) AND N=6 AND  
O(N,1)<>-1 THEN PRINT " THE PIECE  
OF PAPER

HAS WRITTEN ON IT IN AN UNTIDY SCR  
AWL THE NUMBERS ' 5 3 8 1 9 0 0 '.

": RETUR

N

5320 IF S#=0\$(N,2, TO 3) AND N=20 AN  
D O(N,1)=RM THEN PRINT " THE PRINCE  
SS TURNS

AROUND AND SLAPS YOU IN THE FACE!  
": RETURN

5330 IF S#="PAI" AND RM=32 OR RM=38  
THEN PRINT " THE PAINTINGS ARE OF D  
EATH

HIMSELF. AS YOU EXAMINE THE PA  
INTINGS A HOLE OPENS UP IN THE FL  
OOR, YOU

FALL DOWN THE HOLE!": LET RM=18:  
RETURN

5340 IF S#="ALT" AND RM=21 THEN PRI  
NT " YOU EXMINE THE ALTAR. YOU FIND

```

    THAT TH
    ERE IS A CRACK IN IT!": RETURN
5350 IF S$="MEN" AND RM=79 THEN PRI
    NT " YOU EXAMINE THE DRUNKEN MEN.
    ONE OF
    THE MEN REMOVES A KNIFE FROM HIS PO
    CKET AND STABS YOU IN THE HEART. Y
    OU ARE N
    OW VERY DEAD": GO TO 7500
5360 IF S$=0$(10,2, TO 3) AND RM=0(1
    0,1) THEN PRINT " THE BULLET IS SIL
    VER AND
    WILL KILL ANYTHING.": RETURN
5370 NEXT N
5380 GO TO 550
5390 CLS
5395 LET SQ=0
5400 PRINT AT 1,12;"INVENTORY"
5410 PRINT : PRINT : PRINT
5420 FOR N=1 TO IN-1
5430 PRINT " "": IF I$(N,1)<>" TH
    EN PRINT I$(N,2): LET SQ=SQ+1
5440 NEXT N
5450 IF SQ=0 THEN PRINT " YOU HAVE
    NOTHING WITH YOU."
5460 LET SQ=0
5470 RETURN
5480 IF RM=24 AND W<>1 THEN GO SUB
    7490
5490 IF P(RM,1)<>0 THEN LET RM=P(RM
    ,1): PRINT D$(7): RETURN
5500 IF P(RM,1)=0 THEN PRINT D$(2):
    RETURN
5510 IF P(RM,2)<>0 THEN LET RM=P(RM

```

```

,2): PRINT D$(7): RETURN
5520 IF P(RM,2)=0 THEN PRINT D$(2):
RETURN
5530 IF P(RM,3)<>0 THEN LET RM=P(RM
,3): PRINT D$(7): RETURN
5540 IF P(RM,3)=0 THEN PRINT D$(2):
RETURN
5550 IF P(RM,4)<>0 THEN LET RM=P(RM
,4): PRINT D$(7): RETURN
5560 IF P(RM,4)=0 THEN PRINT D$(2):
RETURN
5570 IF P(RM,5)<>0 THEN LET RM=P(RM
,5): PRINT D$(7): RETURN
5580 IF P(RM,5)=0 THEN PRINT D$(2):
RETURN
5590 IF P(RM,6)<>0 THEN LET RM=P(RM
,6): PRINT D$(7): RETURN
5600 IF P(RM,6)=0 THEN PRINT D$(2):
RETURN
5610 IF S$<>"WER" AND S$<>"PRI" THEN
PRINT " I CAN'T SHOOT A ";Q$: RETU
RN
5620 IF S$="WER" AND O(23,1)=RM THE
N GO TO 5640
5631 IF S$="PRI" AND O(20,1)=RM THEN
GO TO 5655
5635 PRINT ,,,," ONE OF US MUST BE M
ISTAKEN. THERE IS NO SUCH CREATU
RE HERE
BUD.": RETURN
5641 FOR N=1 TO IN: IF I$(N,1,3)="GU
N" THEN LET SA=1
5642 IF I$(N,1,3)="BUL" THEN LET SB
=1

```



```

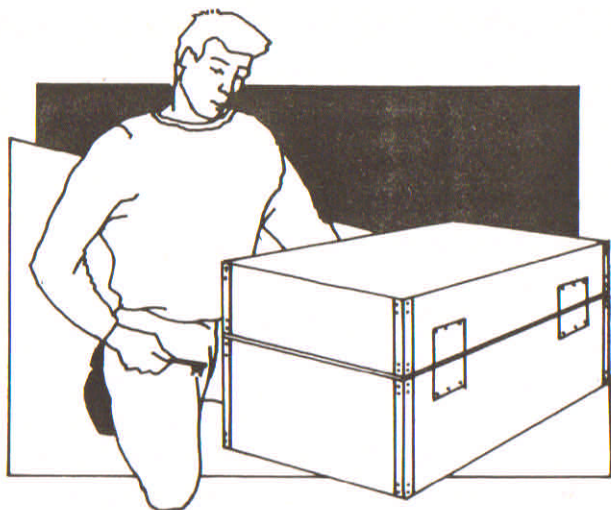
5643 NEXT N
5644 IF SA=1 AND SB=0 THEN PRINT F
LASH 1;" YOU DON'T HAVE A BULLET!";
FLASH 0:
RETURN
5645 IF SA=0 AND SB=0 THEN PRINT "
YOU POINT YOUR TOE AT THE WERE- WOL
F": PRIN
T " BANG! YOUR TOE WASN'T LOADED!":
RETURN
5646 IF SA=1 AND SB=1 THEN PRINT "
YOU POINT YOUR GUN AT THE WERE- WOLF
. BANG!!
IT NOW LIES DEAD ON THE FLOOR.":
LET W=0: RETURN
5656 FOR N=1 TO IN: IF I$(N,1,3)="GU
N" THEN LET SA=1
5657 IF I$(N,1,3)="BUL" THEN LET S
B=1
5658 NEXT N: IF SA=1 AND SB=0 THEN
PRINT " YOU CAN'T SHOOT WITH AN EMPT
Y GUN
CAN YOU ?": RETURN
5659 IF SA=0 AND SB=0 THEN PRINT "
YOU POINT YOUR TOE AT THE PRIN
CESS, BA
NG! YOUR TOE WASN'T LOADED.": RETURN
5660 IF SA=0 AND SB=1 THEN PRINT "
YOU CAN'T SHOOT WITHOUT A GUN CAN
YOU? DUM
MY!": RETURN
5661 IF SA=1 AND SB=1 THEN PRINT "
I DON'T BELIEVE IN VIOLENCE! SO I
WON'T L

```

```

ET YOU SHOOT THE PRINCESS.": RETURN
5700 INK 5: PRINT FLASH 1;" TYPE ON
WHAT? ": INK 7: RETURN
5750 STOP
7490 FOR M=1 TO 50: BEEP .006,M: BORDER
RND*7: NEXT M: BORDER 7: PRINT "
THE WE
REWOLF DRIVES ITS FANGS INTO YOUR
NECK. YOU ARE NOW VERY, VE
RY DEAD!
"
7500 PRINT AT 20,0;" ANOTHER GAME ?"
;: INPUT A$: IF A$="Y" OR A$="YES" T
HEN RUN
7510 REM STOP
7520 CLS : FOR N=1 TO 2000: NEXT N:
PAPER 6
7530 PRINT FLASH 1;" CONGRATULATI
ONS!! YOU HAVE FOUND THE RING
OF POWER
. YOU CAN NOW RULE THE
WORLD!! "
7540 FOR I=1 TO 50 STEP .2: BEEP .00
2,I: NEXT I
7550 STOP

```



```
10 DIM P$(97,192): DIM P(97,6): DI
M O$(40,2,32): DIM O(40,2): DIM W$(1
6,3)
```

```
20 BORDER 0: PAPER 0: INK 7
```

```
30 INK 5: PRINT AT 3,6: FLASH 1;"
The Ring of Power ": INK 2: PRINT AT
5,13: F
```

```
LASH 1;" by ": INK 7: PRINT AT 7,9:
FLASH 1;" Dave Edwards "
```

```
40 FOR n=30 TO 40
```

```
50 CIRCLE 120,60,n
```

```
60 NEXT n
```

```
70 FOR n=0 TO 10
```

```
80 CIRCLE 120,100,n
```

```
90 NEXT n
```

```
100 PRINT FLASH 1;AT 21,9;" Please
Wait "
```

```
180 FOR N=1 TO 97: READ P$(N),P(N,
1),P(N,2),P(N,3),P(N,4),P(N,5),P(N,6
): NEXT
```

```
N
```

```

190 FOR N=1 TO 40: READ O*(N,1),O(N
,1),O*(N,2),O(N,2): NEXT N
200 FOR N=1 TO 16: READ W*(N): NEXT
N
5800 DATA " OUTSIDE THE CASTLE, IT
IS MASSIVE AND AWESOME. YOU HEAR
A CRY O
F HELP, YOU MUST GO FORWARD!",6,0
,0,0,0,0
5810 DATA " IN A DARK MAZE",5,14,14,
3,0,0
5820 DATA " IN A DARK ROOM, AS YOU
ENTER YOU FALL DOWN A DEEP PIT AT T
HE BOTTO
M THERE IS A PILE OF SHARP SPIKES.
YOU LAND ON TOP OF THEM. YOU ARE
VERY VE
RY DEAD!",0,0,0,0,0,0
5830 DATA " IN A DARK MAZE",14,3,5,1
4,0,0
5840 DATA " IN A DARK MAZE",13,2,14,
4,0,0
5850 DATA " INSIDE THE CASTLE. YOU
ARE IN A DAMP, SMELLY, GATE ROOM
",14,0,1
4,5,0,0
5860 DATA " IN A DARK MAZE",14,14,14
,14,0,0
5870 DATA " IN A DARK MAZE",14,10,14
,14,0,0
5880 DATA " IN A DARK MAZE",7,14,3,3
,0,0
5890 DATA " AT THE EXIT IN THE MAZE"
,8,3,3,3,16,0
5900 DATA " IN A DARK MAZE",3,8,3,12
,0,0

```

5910 DATA " IN A DARK MAZE", 3, 3, 11, 1  
5, 0, 0

5920 DATA " IN A DARK MAZE", 14, 15, 15  
, 0, 0, 0

5930 DATA " IN A DARK MAZE", 3, 3, 3, 3,  
0, 0

5940 DATA " IN A DARK MAZE", 3, 6, 12, 1  
3, 0, 0

5950 DATA " ON A SECOND LEVEL  
HALLWAY.", 17, 0, 0, 0, 0, 0

5960 DATA " IN A CORRIDOR OF DOOM.",  
18, 16, 0, 22, 0, 0

5970 DATA " IN A CORRIDOR OF DOOM.  
ALCOVES LEAD OFF TO THE LEFT AND  
RIGHT.",  
21, 17, 20, 19, 0, 0

5980 DATA " IN THE WEST ALCOVE  
THERE ARE WINDOWS ON THE NORTH AND  
SOUTH WALLS, OUT OF THE WINDOWS YOU CAN  
SEE HUNDREDS OF NAKED PEOPLE BEING TORTURED"

, 0, 0, 18, 0, 0, 0

5990 DATA " IN THE EAST ALCOVE  
THERE ARE WINDOWS ON THE NORTH AND  
SOUTH WALLS

OUT OF THE WINDOWS YOU CAN  
SEE HUNDREDS OF NAKED PEOPLE BEING TORTURED"

, 0, 0, 0, 18, 0, 0

6000 DATA " IN AN ALTAR ROOM.", 0, 18,  
0, 0, 0, 0

6010 DATA " IN A LONG CORRIDOR", 0, 0,  
17, 23, 0, 0

6020 DATA " IN A LONG, DAMP, SMELLY

CORRIDOR.", 25, 24, 22, 0, 0, 0  
 6030 DATA " IN THE WEREWOLFS LAIR.",  
 23, 0, 0, 0, 0, 0  
 6040 DATA " IN A LONG, DAMP, SMELLY  
 CORRIDOR.", 26, 23, 0, 0, 0, 0  
 6050 DATA " IN A LIFT SHAFT! AS  
 YOU ENTER YOU FEEL A SUDDEN FALL  
 ING SENS  
 ATION. YOU ARE FALLING DOWN THE  
 SHAFT!", 0, 0, 0, 0, 0, 0  
 6060 DATA " IN A LONG, DAMP, SMELLY  
 CORRIDOR.", 0, 0, 28, 17, 0, 0  
 6070 DATA " IN A LONG, DAMP, SMELLY  
 CORRIDOR.", 29, 0, 0, 27, 0, 0  
 6080 DATA " IN A TORTURE ROOM, YOU  
 SEE HUNDREDS OF NAKED BODIES.", 0, 28,  
 0, 0, 0, 0  
 6090 DATA " IN A STAIR ROOM.", 0, 29, 0  
 , 0, 31, 1  
 6095 DATA " ON THE THIRD STAIR  
 LEVEL", 32, 0, 0, 0, 0, 30  
 6100 DATA " IN A ROOM FULL OF  
 PAINTINGS", 38, 31, 37, 33, 0, 0  
 6110 DATA " ON A STAIR CASE WHICH  
 LEADS UP", 0, 0, 0, 0, 34, 32  
 6120 DATA " IN A ROOM WITH A WELL  
 IN IT. ON THE WEST WALL A SIGN SAYS  
 :-' MOO  
 R FOR HET A DOCE' ON THE NORTH WAL  
 L A SIGN SAYS :- ' THERE IS A W  
 AY TO KE  
 EP THINGS SAFE YOU NEED A ROOM TO  
 FIT.'", 0, 0, 0, 0, 0, 33  
 6130 DATA " IN A PADDED CELL ON THE  
 FLOOR LIES AN OLD MAN.", 0, 0, 34, 0, 0, 0  
 6140 DATA " IN A SAFE ROOM, A SIGN

ON THE WEST WALL SAYS :- ' DE  
POSITE A

LL TREASURES HERE' ", 0, 34, 0, 0, 0, 0

6150 DATA " IN A LONG, DAMP, SMELLY  
CORRIDOR. ON THE EAST WALL LISE A DO  
OR", 40, 1

5, 0, 32, 0, 0

6160 DATA " IN A LONG CORRIDOR  
WHICH HAS PAINTINGS ON EACH WALL  
.", 39, 32

, 0, 0, 0, 0

6170 DATA " IN A GUN ROOM.", 0, 38, 0, 0  
, 0, 0

6180 DATA " IN AN ANTEROOM.", 0, 37, 0,  
0, 0, 0

6190 DATA " IN A COMPUTER ROOM, A  
VDU DISPLAYS A MESSAGE WHICH SAYS  
:- ' EN

TER THE EXIT CODE ' ", 0, 0, 0, 0, 0, 0

6200 DATA " IN AN OPEN STAIR WELL.",  
0, 0, 0, 0, 0, 43

6210 DATA " AT THE ENTRANCE TO A  
SINISTER MAZE. A SIGN SAYS:- ' TH  
E KEY IS

IN THE NOTE' ", 44, 0, 0, 0, 0, 0

6220 DATA " IN THE MAZE", 59, 43, 63, 45  
, 0, 0

6230 DATA " IN THE MAZE", 46, 0, 1, 63, 0  
, 0

6240 DATA " IN THE MAZE", 47, 1, 0, 0, 0,  
0

6250 DATA " IN THE MAZE", 60, 62, 48, 61  
, 0, 0

6260 DATA " IN THE MAZE", 60, 62, 49, 63  
, 0, 0

6270 DATA " IN THE MAZE", 3, 50, 63, 63,  
0, 0

6280 DATA " IN THE MAZE", 49, 51, 63, 60  
, 0, 0

6285 DATA " IN THE MAZE", 50, 50, 52, 0,  
0, 0

6290 DATA " IN THE MAZE", 53, 52, 3, 51,  
0, 0

6300 DATA " IN THE MAZE", 54, 52, 59, 59  
, 0, 0

6310 DATA " IN THE MAZE", 55, 53, 59, 59  
, 0, 0

6320 DATA " IN THE MAZE", 0, 54, 0, 56, 0  
, 0

6330 DATA " IN THE MAZE", 0, 49, 55, 57,  
0, 0

6340 DATA " IN THE MAZE", 60, 60, 56, 58  
, 0, 0

6350 DATA " AT THE EXIT TO THE  
MAZE, STAIRS LEAD UP TO THE NEXT  
LEVEL, "  
, 0, 0, 0, 0, 64, 0

6360 DATA " IN THE MAZE", 1, 1, 1, 1, 0, 0

6370 DATA " IN THE MAZE", 60, 61, 62, 63  
, 0, 0

6380 DATA " IN THE MAZE", 60, 61, 62, 63  
, 0, 0

6390 DATA " IN THE MAZE", 60, 61, 62, 63  
, 0, 0

7000 DATA " IN THE MAZE ", 60, 61, 62, 6  
3, 0, 0

7010 DATA " AT THE TOP OF THE  
STAIRS ", 67, 0, 66, 65, 0, 0

7020 DATA " IN THE KITCHEN ", 0, 0, 64,  
0, 0, 0

7030 DATA " IN THE PANTRY ", 0, 0, 0, 64



,0,0

7040 DATA " IN A HALLWAY LEADING  
TO THE LIVING ROOM",69,64,0,0,0,0

7050 DATA " IN THE LIVING ROOM. A  
SIGN ON THE NORTH DOOR SAYS :- HERE  
WITHIN

LIES THE RING OF POWER DO YOU WIS  
H TO TAKE IT ?",0,67,70,69,0,0

7060 DATA " IN A HALLWAY LEADING  
TO THE DINING HALLS",71,0,68,0,0,0

7070 DATA " IN A HALLWAY LEADING  
TO THE DINING HALLS",73,0,0,69,0,0

7080 DATA " IN THE DINING HALLS",78,  
69,0,0,0,0

7090 DATA " IN THE RING ROOM ",0,68,  
0,0,0,0

7100 DATA " IN THE DINING HALLS ",78  
,70,0,0,0,0

7110 DATA " IN A SMOKING ROOM",0,71,  
75,0,0,0

7120 DATA " IN A BEDROOM ",0,0,76,74  
,0,0

7130 DATA " IN A CONNECTING HALL-  
-WAY ",79,0,77,75,0,0

7140 DATA " IN A STORE ROOM",0,0,78,  
0,0,0

7150 DATA " IN A LARGE BEDROOM",0,73  
,0,0,0,0

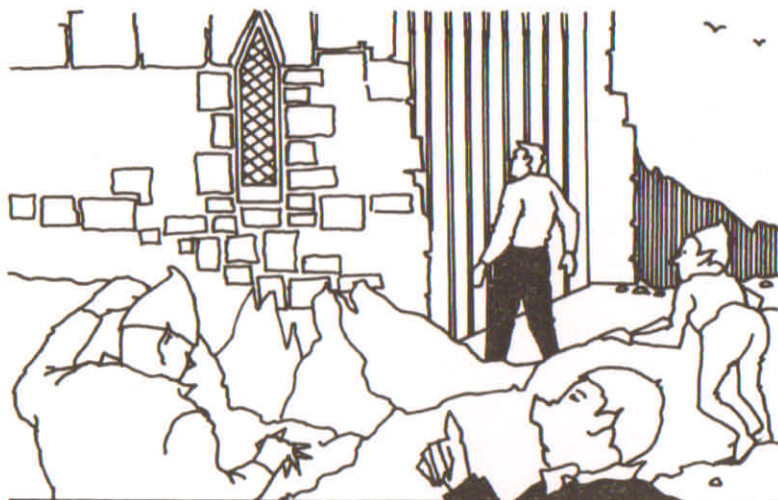
7160 DATA " IN AN ANTEROOM FULL OF  
DRUNKEN MEN",1,76,0,0,0,0

7170 DATA " IN A SWITCH ROOM",0,0,43  
,0,0,0

7180 DATA " IN A CONNECTING  
CORRIDOR",0,0,82,37,0,0

7190 DATA " IN AN ALTAR ROOM

DEVOTED TO LIFE", 0, 0, 83, 81, 0, 0  
 7200 DATA " IN A ROOM WITH A FISH  
 POND AT THE EAST END, THE POND HAS  
 LOTS OF  
 CARP SWIMMING AROUND IN IT", 0,  
 0, 0, 82, 0, 0



7210 DATA " IN THE GARDENERS STORE "  
 , 0, 86, 85, 83, 0, 0  
 7220 DATA " IN THE GARDENERS STORE "  
 , 0, 87, 0, 84, 0, 0  
 7230 DATA " IN THE GARDENERS STORE "  
 , 84, 0, 87, 0, 0, 0  
 7240 DATA " IN THE GARDENERS STORE "  
 , 85, 0, 0, 86, 0, 0  
 7250 DATA " IN A CONNECTING  
 CORRIDOR , WHICH IS SMELLY AND DAMP  
 " , 82, 89,  
 0, 0, 0, 0

7260 DATA " IN A CONNECTING  
CORRIDOR , WHICH IS SMELLY AND DAMP  
. AN ALT

AR LIES AT THE SOUTH END WHICH  
IS DEVOTED TO BIRTH", 88, 0, 0, 0, 0, 0

7270 DATA " IN A SMALL, DARK,  
PASSAGE", 0, 91, 93, 0, 0, 0

7280 DATA " IN A SMALL, DARK,  
PASSAGE", 90, 92, 0, 0, 0, 0

7290 DATA " IN A SMALL, DARK,  
PASSAGE. THERE IS A STATUE OF A REAP  
ER AT TH

E EAST END.", 91, 0, 0, 0, 0, 0

7300 DATA " IN A SMALL, DARK,  
PASSAGE.", 89, 0, 94, 90, 0, 0

7310 DATA " IN A SMALL, DARK,  
PASSAGE.", 0, 95, 0, 93, 0, 0

7320 DATA " IN A SMALL, DARK,  
PASSAGE.", 94, 96, 0, 0, 0, 0

7330 DATA " IN A SMALL, DARK,  
PASSAGE. THERE IS A VAMPIRE ON THE  
WEST WAL

L", 95, 0, 0, 0, 0, 0

7340 DATA " IN THE STORE ROOM OF  
THE MINDSTONE", 0, 0, 96, 0, 0, 0

7350 REM

7360 REM

7370 DATA "A BONE", 17, "BON", 1, "A BIB  
LE", 19, "BIB", 2, "A KEY", -2, "KEY", 3, "A  
BAG OF

COINS", 25, "COI", 4, "A DAGGER", 20, "DAG  
, 5, "A PIECE OF PAPER", 35, "PAP", 6, "T  
HE MINDS

TONE", 97, "MIN", 7, "A KITCHEN KNIFE", 6  
5, "KNI", 8, "A WOODEN CROSS", 66, "CRO",

9, "A SIL  
VER BULLET", 69, "BUL", 10  
7380 DATA "A PILE OF FOOD", 71, "FOO",  
11  
7390 DATA "SOME WATER", 76, "WAT", 12, "  
A BARREL OF BEER", 77, "BEE", 13, "A BOT  
TLE", 73,  
"BOT", 14, "THE RING OF POWER", 72, "RIN  
", 15, "A GUN", 39, "GUN", 16  
7400 DATA "A DIAMOND", 22, "DIA", 17, "A  
GOLD BAR", 23, "BAR", 18, "A SILVER BAR  
", 67, "SI  
L", 19, "A BEAUTIFUL PRINCESS", 27, "PRI  
", 20, "A SWITCH", 80, "", 21, "A DOG", 40,  
"", 22, "A  
WEREWOLF", 24, "", 23, "A GOLDEN KEY", 3  
6, "GOL", 24, "A CARROT", 15, "CAR", 25  
7410 DATA "AN IRON KEY", 18, "IRO", 26,  
"A PLATINUM KEY", 38, "PLA", 27, "A JAR  
OF HONEY  
", 10, "HON", 28, "A MAGIC SPELL BOOK", 1  
4, "BOO", 29, "A SPADE", 84, "SPA", 30, "FO  
RK", 87, "  
FOR", 31, "A SWORD", 90, "SWO", 32, "A CAN  
DLE", 76, "CAN", 33, "A PILE OF SHEETS",  
75, "SHE"  
, 34, "A ROD"  
7420 DATA 61, "ROD", 35, "A MAGIC WAND"  
, 50, "WAN", 36, "A TUBE OF TOOTHPASTE",  
59, "TUB"  
, 37, "A BOOK ON D.I.Y CASTLES", 63, "DI  
Y", 38, "A PEN", 55, "PEN", 39, "A ROLL OF  
PAPER",  
57, "ROL", 40  
7430 REM  
7440 DATA "GET", "TAK", "DRO", "KIL", "U

NL", "EXA", "LOO", "INV", "SHO", "TYP", "N  
OR", "SOU

", "EAS", "WES", "UP", "DOW"

7450 REM

7460 REM

7470 REM

7480 CLS

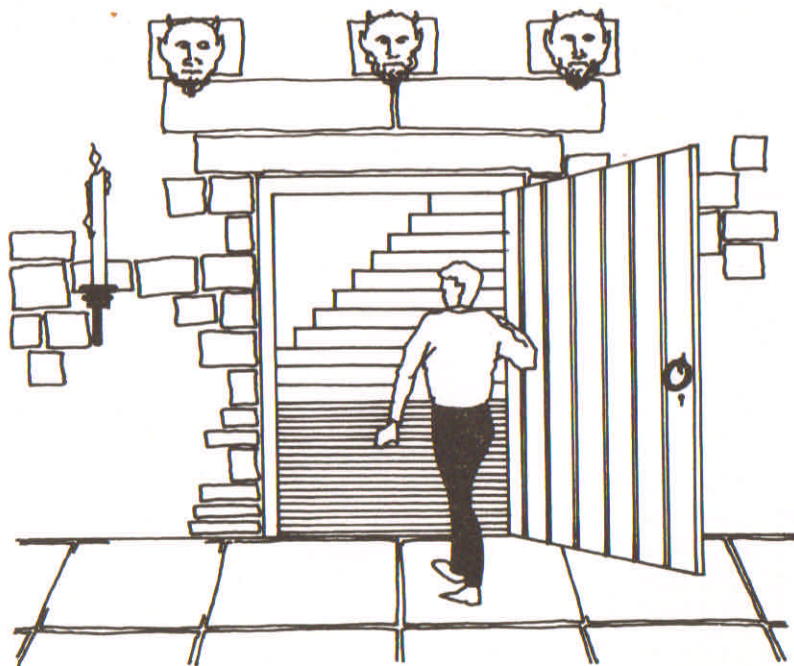
7500 SAVE "1" DATA p\$( )

7510 SAVE "2" DATA p( )

7520 SAVE "3" DATA o\$( )

7530 SAVE "4" DATA o( )

7540 SAVE "5" DATA w\$( )



# THE SEVEN KEYS OF TARKUS

In the windswept lands of Asgard north of the great city of Gamroth lies the vast system of chambers, caves and tunnels known as Tarkus. This system was originally a plaything for the wife of Errung the Invincible; however, with the star god's great demise, the system fell into disuse. Legends abound concerning the system and its contents, but one thing is clear — the soul of the mischievous god, Loki, is hidden there in the Inner Chamber. Seven keys are needed to open the casket that contains Loki's poor tormented soul.

You were foolish and your thoughts were unclear at the time but you consented to the gods' wishes and have pledged to release Loki's soul. After all, what other way could you pay back your enormous bar bill when you launched a huge party at the Mega-Galactic Zeus Tearooms?

You are now in the first chamber of Tarkus after a long and uneventful trip across the desolate Northern Plain. Good luck, you must not fail.

Here are a couple of hints for you as you tackle this tough adventure, utilising the high resolution graphics of your computer. Firstly, try and map your path through the chambers; going about the game in a haphazard manner will get you nowhere. Secondly, watch out for dead ends which might trap you.

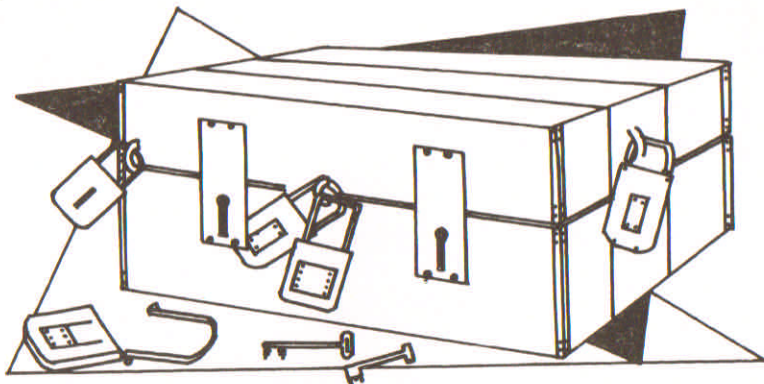
0>CLS

10 BORDER 7: PAPER 7: INK 0: PRINT  
AT 10,10;"data loading"

```

20 GO SUB 1160: REM *U*D*G*S*
30 GO SUB 0390: GO SUB 0950
40 LET k=0
50 LET l=83
60 LET n=0: LET s=0: LET e=0: LET
w=0
70 CLS : GO SUB 1210
80 REM ===Main===Loop=====
90 LET dc=0
100 INK 1
110 IF a(1,1)=1 THEN LET n=1: GO S
UB 1310
120 IF a(1,3)=1 THEN LET w=1: GO S
UB 1420
130 IF a(1,4)=1 THEN LET e=1: GO S
UB 1380
140 IF a(1,2)=1 THEN LET s=1: GO S
UB 1460

```



```

150 INK 0: PRINT AT 21,4;" Number o
f Keys:- ";k;
160 IF l=17 OR l=37 OR l=67 OR l=11
OR l=82 THEN PRINT AT 14,10; FLASH
1;"

```

```

DEAD  ";; PRINT AT 15,10; FLASH 1;"
      END.  ";; GO TO 0160
170 IF k=7 AND l=27 THEN GO TO 070
0
180 FOR x=1 TO 100: IF l=x AND k(x)
=1 THEN LET dc=1: BORDER 0: BORDER
7: PRINT
  AT 12,14; FLASH 1;"  ";; PRINT AT
  13,14; FLASH 1;"  ";; LET x=100:
NEXT x: GO TO 0200
190 NEXT x
200 IF dc=1 THEN LET a$=INKEY$: IF
a$="" THEN GO TO 0200
210 IF dc=1 THEN IF INKEY$="t"?THE
N LET k=k+1: LET k(1)=0: BEEP .02,2
3: BEEP
.06,30: BEEP .08,36: PRINT AT 12,14;
"  ";; PRINT AT 13,14;"  ";; GO
TO 0380
220 IF INKEY$="" THEN GO TO 0220
230 IF INKEY$="q" AND l>10 AND a(1,
1)=1 THEN GO SUB 0310: LET l=l-10:
GO TO 03
80
240 IF INKEY$="a" AND a(1,2)=1 AND
l<90 THEN GO SUB 0310: LET l=l+10:
GO TO 03
80
250 IF INKEY$="o" AND a(1,4)=1 THEN
GO SUB 0310: LET l=l-1: GO TO 0380
260 IF INKEY$="p" AND a(1,3)=1 THEN
GO SUB 0310: LET l=l+1: GO TO 0380
270 IF dc=1 AND INKEY$="t" THEN LE
T k=k+1: LET a(1,1)=0: LET a(1,2)=0:
LET a(1

```



```

,3)=0: LET a(1,4)=0: GO TO 0380
280 REM ***wrong way silly****
290 BEEP .05,10: BEEP .1,1: BEEP .1
1,10
300 FOR t=1 TO 500: NEXT t: GO TO 0
210
310 OVER 1
320 IF s=1 THEN GO SUB 1460: LET s
=0
330 IF e=1 THEN GO SUB 1380: LET e
=0
340 IF w=1 THEN GO SUB 1420: LET w
=0
350 IF n=1 THEN GO SUB 1310: LET n
=0
360 OVER 0
370 RETURN
380 FOR i=1 TO 10: BEEP .04,i: NEXT
i: GO TO 0080
390 REM ***Initialisation***
400 DIM a(100,4): DIM k(100)
410 FOR x=1 TO 100
420 FOR y=1 TO 4: READ a(x,y): NEXT
y
430 NEXT x
440 FOR x=1 TO 100: READ k(x): NEXT
x
450 REM ***Data***
460 DATA 0,0,0,0,63,63,63,63,1,3,7,
15,255,255,255,255,60,60,60,60,60,60
,0,0,207
,207,207,15,15,15,0,0,255,15,7,3,0,0
,0,0,0
470 DATA 0,0,1,0,0,1,1,1,0,0,1,1,0,
0,0,1,0,1,0,0

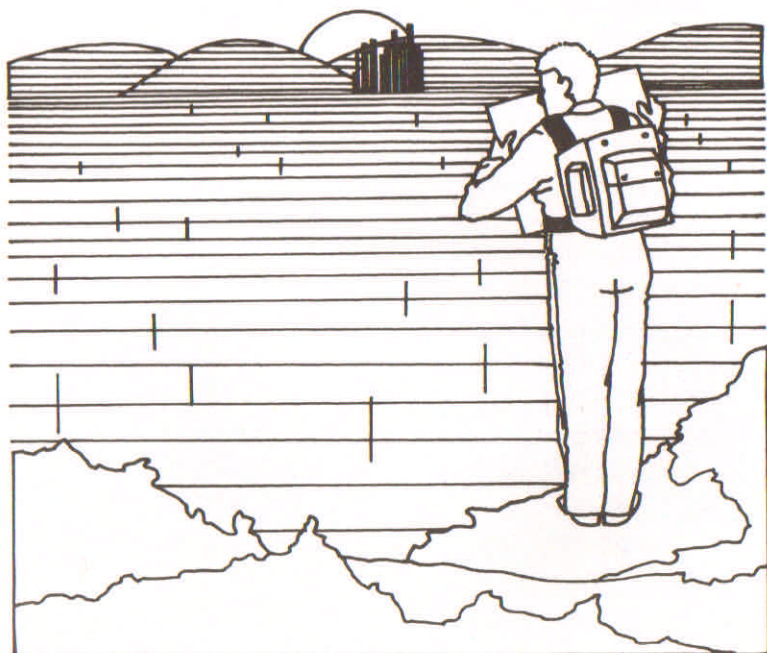
```

ADVENTURES FOR YOUR  
ZX SPECTRUM

```

480 DATA 0,1,1,0,0,0,1,1,0,1,0,1,0,
1,1,0,0,1,0,1
490 DATA 0,0,0,0,1,1,1,0,0,0,1,1,0,
0,0,1,1,1,1,0
500 DATA 1,1,1,1,0,0,0,0,1,1,1,0,1,
1,0,1,1,1,0,0
510 DATA 1,1,1,0,1,0,0,1,0,0,1,0,0,
0,0,1,1,0,0,0
520 DATA 1,1,0,0,0,0,1,0,1,0,0,1,1,
1,0,0,1,0,0,0
530 DATA 1,0,1,0,0,1,1,1,0,0,1,1,0,
0,1,1,0,0,1,1
540 DATA 1,1,1,1,0,0,1,1,0,0,1,1,1,
1,1,1,0,1,0,1
550 DATA 0,1,0,0,1,1,1,0,0,1,1,1,0,
1,0,0,0,0,1,0

```



```

560 DATA 1,1,0,1,0,1,1,0,0,0,0,1,1,
1,0,0,1,0,0,0
570 DATA 1,1,1,0,1,0,0,1,1,1,0,0,1,
1,0,0,0,1,0,0
580 DATA 1,1,1,0,1,1,1,1,0,0,0,1,1,
0,1,0,0,1,0,1
590 DATA 1,1,1,0,0,0,0,1,1,1,1,0,1,
0,1,1,1,0,0,1
600 DATA 1,1,1,0,1,1,1,1,0,0,1,1,0,
0,1,1,1,1,0,1
610 DATA 1,1,1,0,0,0,1,1,1,1,1,1,0,
1,1,1,0,1,1,1
620 DATA 1,1,0,1,1,0,1,0,0,0,1,1,0,
1,1,1,1,0,0,1
630 DATA 1,1,0,0,0,0,0,0,1,0,0,0,1,
0,1,0,1,1,0,1
640 DATA 1,1,0,0,0,1,1,0,0,0,1,1,1,
0,1,1,0,1,0,1
650 DATA 1,0,0,0,1,0,1,0,0,0,1,1,0,
0,1,1,1,0,0,1
660 DATA 1,0,1,0,1,0,0,1,0,0,1,0,0,
0,0,1,1,0,0,0
670 DATA 1,0,0,0,0,0,0,0,0,1,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1
680 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,
0,0,0,0,
0,0,0,1,0,0,0,0,0,0,0,0,1
690 RETURN
700 REM ***The inner chamber***
710 LET w=INT (RND*50): LET d=INT (
RND*20): IF d<4 OR w<10 THEN GO TO
0710

```

```

720 FOR i=1 TO 16 STEP .2: BEEP .00
8,-1+i: BEEP .005,2+i: BEEP .005,4+i
: BEEP .
05,5+i: NEXT i
730 CLS : PRINT : PRINT " Well don
e! you have the seven keys and have
reached
the inner sanctum of Tarkus...."
740 PRINT
750 PRINT " However, there is one
task to still complete before Loki's
soul
can be released."
760 PRINT
770 PRINT " The guardians of Tark
us, now long dead, have left a water
valv
e on the box containing Loki's s
oul."
780 FOR t=1 TO 2000: NEXT t: CLS :
PRINT : PRINT
790 PRINT " The valve needs ";w;"
Litres of water to open it": PRINT
"But the
only medium available is sand and th
at weighs ";d;" times less than wa
ter"
810 PRINT "You must calculate how mu
ch is needed and quickly, you only
have
30 seconds"
820 PRINT ^0;" Press any key when y
ou've found the answer
"
830 LET ans=w*d

```

```

840 FOR i=1 TO 2290: IF INKEY$>=CHR
$ (48) THEN GO TO 0870: NEXT i
850 NEXT i
860 GO TO 1110
870 CLS : BORDER 0: INPUT " What's
your answer then? ";a
880 IF a<>ans THEN GO TO 1110
890 CLS : PRINT ^0;"****Welldone my
friend your*****debt is paid. L
oki's***
*****soul is free at last.*****
*****The God's are most*****
*****g
rateful*****"
900 FOR i=0 TO 7: BORDER i: RANDOMI
ZE USR 3280: NEXT i
910 FOR i=1 TO 1000
920 NEXT i
930 FOR i=0 TO 60: RANDOMIZE USR 32
80: BEEP (i/500),i: NEXT i
940 STOP
950 REM **Instructions**
960 INK 0: CLS : PRINT : PRINT
970 PRINT FLASH 1;" THE SEVEN K
EYS OF TARKUS "
980 PRINT : PRINT : PRINT " In the
windswept lands of Asgard, nort
h of the
Great City Of Gamroth lies the sy
stem of chambers and caves known as
"; FLASH 1;"TARKUS"
990 PRINT : PRINT " Within the sys
tem, as yet unexplored by any cr
eature a

```

```

re fabled to lie seven mythical
keys. If collected together in the
Inner Sa
nctum of Tarkus, these keys will
unlock ";
1000 PRINT "the soul of Loki The God
Of Light"
1010 PRINT ^0; " Press any
Key "
1020 IF INKEY$(">") THEN GO TO 1040
1030 GO TO 1020
1040 CLS : PRINT ^0; FLASH 1; " TH
E SEVEN KEYS OF TARKUS "
1050 PRINT ^0; PRINT ^0; PRINT ^0; "
With yourself in the debt of the g
reat God
s of the Northern Plain, you find n
o chance but ";
1060 PRINT ^0; "to rescue Loki's soul.

```

You have trekked to the entrance of Tarkus and have just entered the first chamber

```

1070 PRINT ^0; PRINT ^0; " Nothing ca
n ease your fear save for the knowled
ge of th
e keys to use...."

```

```

1080 PRINT ^0; PRINT ^0; " To Move: -"
: PRINT ^0; PRINT ^0; " ""Q""=North
""T
""=take key ""A""=South
""O""=West

```

```

""P""=East"

```

THE SEVEN KEYS OF TARKUS

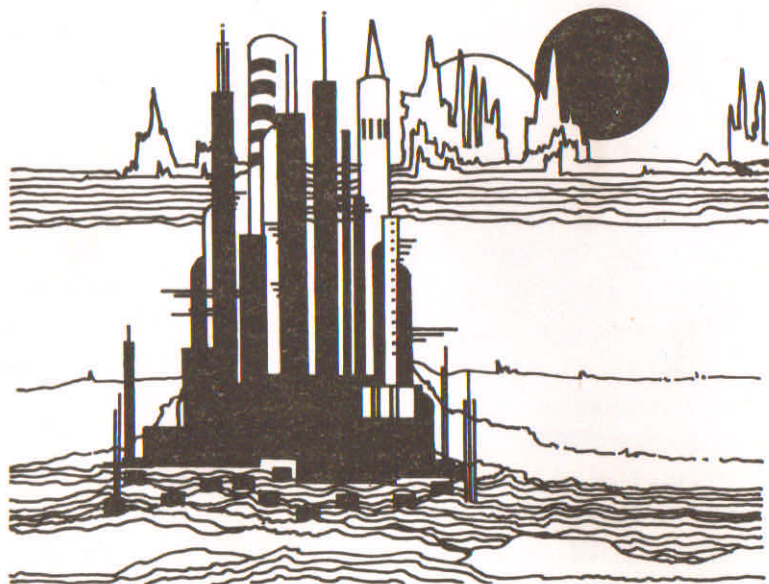
```

1090 IF INKEY$<>" " THEN RETURN
1100 GO TO 1090
1110 REM ***Useless***
1120 PAPER 0: INK 7: CLS : PRINT ^0;
"You Failed To Solve The Final  Pro
blem On
The Valve.                The Answer Was
";ans
1130 FOR i=0 TO 5: BORDER 0: RANDOMI
ZE USR 3280: NEXT i
1140 FOR i=8 TO 0 STEP -1: BEEP .05,
i: NEXT i
1150 STOP
1160 REM ***U***D***G's***
1170 FOR i=0 TO 40: READ user: POKE
USR "a"+i,user
1180 NEXT i
1190 RETURN
1200 REM ****West Door*****
1210 CLS : INK 0
1220 PLOT 0,30
1230 DRAW 255,0,-1.5
1240 PLOT 0,31
1250 DRAW 255,0,-1.5
1260 GO TO 1270: REM ***roof***
1270 FOR i=1 TO 125
1280 INK 1: PLOT i,175: DRAW 255-(i+
i),0,1
1290 NEXT i
1300 RETURN
1310 INK 4: REM ****North door****
1320 PLOT 103,80
1330 DRAW 25,50,-1.5
1340 DRAW 25,-50,-1.5
1350 PLOT 105,80: DRAW 23,49,-1.5

```

```

1360 DRAW 23,-49,-1.5
1370 RETURN
1380 INK 1: REM ****west Door***
1390 PLOT 0,105: DRAW 25,-55,-1.5
1400 PLOT 0,104: DRAW 23,-55,-1.5
1410 RETURN
1420 INK 1: REM ****east Door*****
1430 PLOT 255,105: DRAW -25,-55,1.5
1440 PLOT 255,104: DRAW -23,-55,1.5
1450 RETURN
1460 INK 3: PRINT AT 14,10;"There is
      an";: PRINT AT 15,10;"Exit South";
1470 RETURN
    
```





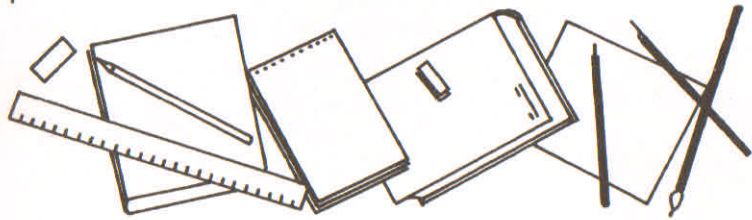
# SCHOOL'S OUT

You've been naughty again. You really should not have put itching powder on the teacher's chair and you are now suffering the consequences...detention after school.

You have never stayed late at school before. Normally as soon as the bell goes you rush out as quickly as possible. If you had stayed late before you would have realised that your school is no ordinary school. After school, things start to get sinister, familiar objects turn nasty and life gets extremely dangerous. You must escape quickly, but how? All I can tell you is that there are 15 commands at your beck and call, which are listed below:

take	read	n = north
get	unlock	s = south
drop	open	e = east
file	kill	w = west
tie	l = look	i = inventory

All commands must be entered in lower case letters. This great adventure, written by Adam Waring, shows how an interesting and exciting adventure can be created by using fairly standard adventure techniques — interlocking rooms, obstructions, and so on — but in a novel scenario such as the school. There are many other possible environments where an adventure might be set, and some are outlined in the chapter on adventure writing.



```

10 GO SUB 8000
20 LET ln=23
30 BORDER 0: PAPER 0: INK 7
90 CLS
100 PRINT "I am in the ";l$(ln)
110 PRINT "Exits are :"
120 FOR i=1 TO 3
130 LET f$=e$(ln,i)
140 PRINT "  ":"North" AND f$="n";
East" AND f$="e";"South" AND f$="s";
"West" A
ND f$="w";
150 NEXT i
160 LET f=1: PRINT "I can see ":"
FOR i=1 TO obj: IF p(i)=ln THEN PRI
NT "  A"
;o$(i,4 TO ): LET f=0
170 NEXT i: IF f THEN PRINT " not
hing at all"
174 IF k=1 THEN GO TO 6800
175 IF ln=12 THEN LET k=k+1
180 INPUT "> "; LINE r$: PRINT ">
";r$
185 IF r$="" THEN GO TO 180
190 IF LEN r$>6 THEN GO TO 300
191 LET r$=r$(1)
195 IF r$="i" THEN GO TO 7000
196 IF r$="l" THEN GO TO 100
198 LET x$="nesw": FOR i=1 TO 4: IF
x$(i)=r$ THEN GO TO 200
199 NEXT i: GO TO 6070
200 FOR i=1 TO LEN e$(ln)
210 IF e$(ln,i)=r$ THEN LET ln=VAL
d$(ln,i*2-1 TO i*2): GO TO 90
220 NEXT i

```

```

230 PRINT "I cant go in that direct
ion": GO TO 100
300 FOR i=4 TO LEN r$
310 IF r$(i)=" " THEN GO TO 340
320 NEXT i
330 GO TO 6070
340 LET x$=r$( TO 3)
350 LET y$=r$(i+1 TO i+3)
360 FOR v=1 TO LEN v$ STEP 3: REM v
erbs
370 IF x$=v$(v TO v+2) THEN GO TO
400
380 NEXT v
390 GO TO 6070
400 FOR n=1 TO obj
410 IF y$=o$(n, TO 3) THEN GO TO 4
40
420 NEXT n
430 PRINT r$( TO i); "what?": GO TO
180
440 LET v=(v-1)/3
450 IF ln=15 AND n=7 AND o$(7,5)="t
" THEN GO TO 6520
490 GO TO 900+v*100
1020 IF p(n)=0 THEN GO TO 6010
1030 IF p(n)<>ln THEN GO TO 6020
1040 IF c>2 THEN GO TO 6030
1050 IF n>9 THEN GO TO 6040
1070 LET c=c+1: LET p(n)=0
1090 GO TO 6000
1110 IF p(n)<>0 THEN GO TO 6050
1120 IF n=6 AND ln=15 THEN GO TO 65
00
1130 IF n=7 AND ln=8 THEN GO TO 651
0

```

```

1170 LET p(n)=1n: LET c=c-1
1190 GO TO 6000
1210 IF p(4)<>0 THEN LET r$=r$( TO
i)+"hacksaw": GO TO 6050
1220 IF n<>11 THEN GO TO 6040
1230 IF 1n<>23 THEN GO TO 6020
1250 IF e$(23)="ne" THEN GO TO 6060
1270 LET o$(11)="win window frame":
LET e$(23)="ne": LET d$(23)="1524"
1290 GO TO 6000
1310 IF p(n)>0 THEN GO TO 6050
1320 IF n<>3 THEN GO TO 6040
1350 INPUT "Tie elastic to > "; LINE
x$
1353 PRINT "> to ";x$
1355 IF LEN x$<3 THEN GO TO 6070
1360 LET r$=r$( TO i)+"elastic to th
e "+x$
1370 IF x$(1 TO 3)<>"sti" THEN GO T
O 6040
1380 LET c=c-1: LET o$(3)="cat catap
ult": LET p(2)=99
1390 GO TO 6000
1410 IF 1n=9 AND n=13 THEN GO TO 65
60
1420 IF 1n=33 AND n=14 THEN GO TO 6
570
1440 IF p(n)>0 THEN GO TO 6050
1450 IF n=8 THEN GO TO 6200
1460 IF n<>9 THEN GO TO 6040
1490 GO TO 6550
1510 IF n<>10 AND n<>15 THEN GO TO
6040
1520 IF NOT (1n=4 AND n=10 OR 1n=20
AND n=15) THEN GO TO 6020

```

```
1525 IF o$(n,7)="n" OR o$(n,7)="p" T
HEN GO TO 6060
1530 IF ln=20 THEN GO TO 1570
1540 INPUT "Safe combination > ";x:
PRINT "> combination : ";x
1550 IF x<>27446+801239+23*48 THEN
GO TO 6600
1560 LET o$(10)="safn unlocked safe"
: GO TO 6000
1570 IF p(5)>0 THEN GO TO 6620
1580 LET o$(15)="doon unlocked door"
1590 GO TO 6000
1610 IF n<>10 AND n<>15 THEN GO TO
6040
1620 IF NOT (ln=4 AND n=10 OR ln=20
AND n=15) THEN GO TO 6020
1630 IF o$(n,7)="p" THEN GO TO 6060
1640 IF o$(n,6)<>"u" THEN LET r$=r$
+" yet": GO TO 6040
1650 IF ln=4 AND n=10 THEN LET o$(1
0)="safn opened safe": LET p(5)=4: G
O TO 600
0
1660 IF ln=20 AND n=15 THEN GO TO 6
700
1690 GO TO 6020
1705 IF n<>16 THEN GO TO 6040
1710 IF ln<>12 THEN GO TO 6020
1720 IF p(1)>0 THEN LET r$=r$( TO i
)+"pellet": GO TO 6050
1730 IF p(3)>0 THEN LET r$=r$( TO i
)+"catapult": GO TO 6050
1740 LET o$(16)="car dead caretaker"
: LET p(1)=99
1750 LET k=2
```

```

1790 GO TO 6650
5010 PRINT "Press a key for another
game"
5020 PAUSE 0
5030 RUN
6000 PRINT "OK..": GO TO 180
6010 PRINT "I'm already carrying th
e";r$(i TO );"!": GO TO 180
6020 PRINT "I dont see the";r$(i TO
);" here!": GO TO 180
6030 PRINT "I'm carrying too much!":
GO TO 180
6040 PRINT "It's impossible to ";r$(
TO i);"the";r$(i TO );"!": GO TO 18
0
6050 PRINT "I'm not carrying the";r$(
i TO );"!": GO TO 180
6060 PRINT "It's allready been done
mate!": GO TO 180
6070 PRINT "I dont know what you're
on about": GO TO 180
6200 PRINT "The poem reads:" "To sol
ve the riddle, here's the key-" "Col
lect tog
ether numbers three," "Add them up a
nd you will see, Your answer gives
you saf
ety.";TAB 20;"Adam Waring."
6205 GO TO 6000
6500 PRINT "The guineapig drinks the
beer!": LET o$(7)="gui drunk guinea
pig": LE
T o$(6)="glan empty beer glass": GO
TO 1130
6510 PRINT "The guineapig staggers i

```

```

nto the laser beam! BANG!": LET o$(7
)="gui f
ried guineapig": LET o$(12)="hol hol
e in the wall": LET e$(8)="esw": LET
d$(8)="
091607": GO TO 1140
6520 PRINT "The guinea pig goes for
your throat and quenches its thir
st on y
our blood! You are dead!!": GO TO 50
00
6550 PRINT "The printout reads :""
'^27446'" "It suddenly self-destructs
in a
puff of smoke!": LET c=c-1: LET p(9)
=99: GO TO 180
6560 PRINT "The blackboard reads: ""
23"" " x48"" " ____"" " ____": GO T
O 180
6570 PRINT "The grafitti reads: ""
If you want a good time, phone801239
and ask
for WYLF!": GO TO 180
6600 PRINT "Wrong number! The alarm
goes offand you are imprisoned for 1
ife and
die in your cell!": GO TO 5000
6620 PRINT "I havn't got anything to
unlock it with!": GO TO 180
6650 PRINT "I fire the pellet from t
he cattyand get him between the eyes
": LET p
(1)=99: LET c=c-1: LET o$(16)="car d
ead caretaker": GO TO 6000
6700 PRINT FLASH 1;"You have escape

```

```

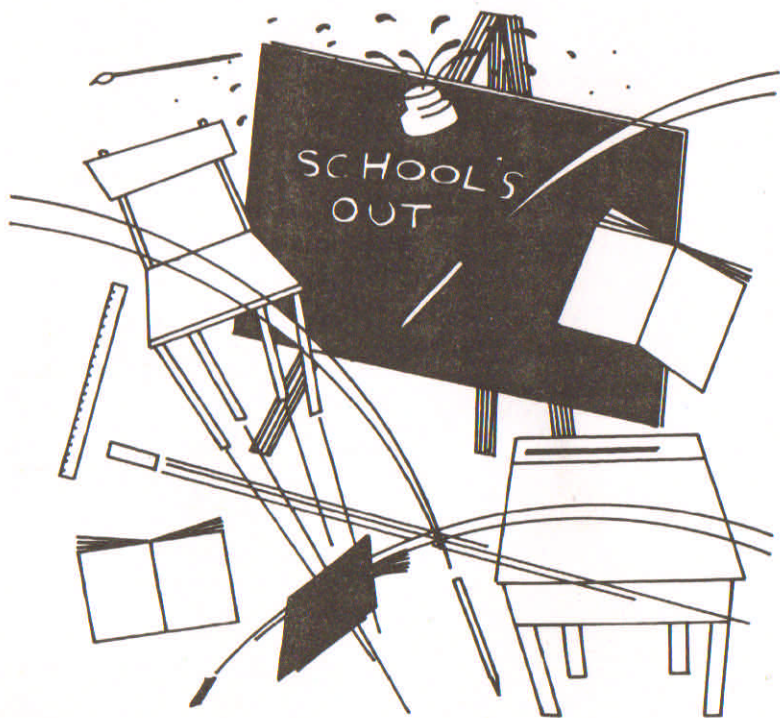
d and won this  adventure!": GO TO
5000
6800 PRINT "The mad caretaker puts a
  broom through your head! You are d
ead!": G
O TO 5000
7005 LET f=1
7010 PRINT "I am carrying:"
7020 FOR i=1 TO obj
7030 IF p(i)=0 THEN PRINT "  A";o$(
i,4 TO ): LET f=0
7040 NEXT i
7050 IF f THEN PRINT "  nothing at
all"
7090 GO TO 180
7999 STOP
8010 LET v$="takgetdrofiltiere aunlop
ekil"
8050 LET obj=16: LET room=34
8060 LET c=0
8070 LET k=0
8100 DIM L$(room,19): DIM e$(room,3)
: DIM d$(room,6)
8130 DIM o$(obj,21): DIM p(obj)
8150 FOR n=1 TO room
8160 READ l$(n),e$(n),d$(n)
8170 NEXT n
8200 FOR i=1 TO obj
8210 READ o$(i),p(i)
8220 NEXT i
8990 RETURN
9010 DATA "Economics room","es","020
6","Geography room","sw","0701","Sta
ff room"
,"es","0410","Headmasters office","e

```



w", "0503", "Secretaries office", "sw",  
 "1204"  
 9020 DATA "History room", "ns", "0114"  
 , "Chemistry lab", "ne", "0208", "Physic  
 s lab", "  
 sw", "1607", "Maths room", "esw", "10170  
 8", "Store room", "nsw", "031809"  
 9030 DATA "Canteen", "es", "1219", "Ass  
 embly hall", "nsw", "052011", "R.E. roo  
 m", "es",  
 "1421", "French room", "new", "061513",  
 "Biology lab", "esw", "162314"  
 9040 DATA "General Science lab", "nw"  
 , "0815", "Computer room", "ns", "0925",  
 "English  
 room", "ns", "1026", "Sick room", "ns",  
 "1127", "Entrance hall", "n", "12"  
 9050 DATA "Latin room", "ne", "1322", "  
 German room", "w", "21", "Detention roo  
 m", "e", "  
 24", "Tech drawing room", "sw", "3023",  
 "Music room", "nes", "172631"  
 9060 DATA "Drama room", "new", "182725  
 ", "Needlework room", "new", "192826", "  
 Art room  
 ", "sw", "3427", "Metalwork room", "e", "  
 30", "Woodwork room", "nw", "2429"  
 9070 DATA "Boys changing room", "ne",  
 "2532", "Gymnasium", "ew", "3331", "Girl  
 s changi  
 ng room", "ew", "3432", "Pottery room",  
 "nw", "2833"  
 9110 DATA "pel ceramic pellet", 34, "s  
 ti 'Y' shaped stick", 30, "ela piece o  
 f elasti

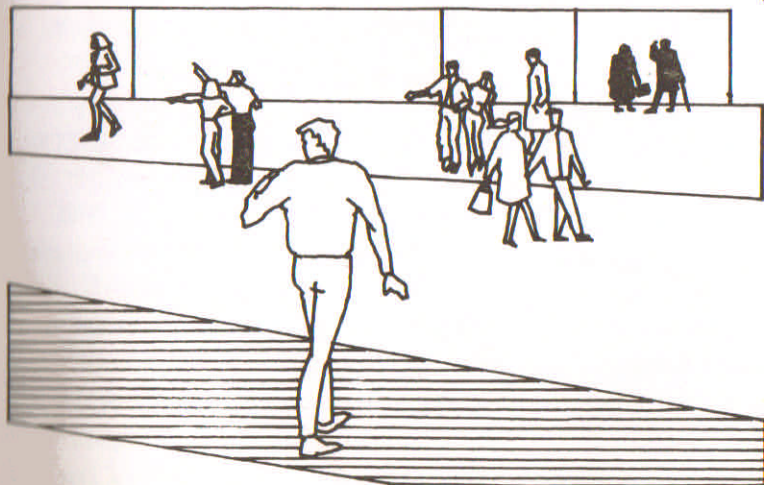
c",27,"fil file",29,"key key",99,"be  
e glass of beer",22,"gui thirsty gui  
neapig",  
15,"poe poem",18  
9120 DATA "pri computer printout",17  
,"saf locked safe",4,"win barred win  
dow",23,  
"las laser beam",8  
9160 DATA "bla blackboard",9,"gra gr  
afittied wall",33,"doo locked main d  
oor",20,  
"car mad caretaker",12



# EVERYDAY ADVENTURE

The title sums up the scenario of this adventure. You start in your house and can then travel round your home streets and hitch a bus ride to the town centre. But I am not going to tell you the aim of the adventure. You must find that out for yourself. Remember, however, that it is an *everyday* situation, wizards will not appear and you will not have to do battle with fearsome monsters. You should try to avoid the bully.

The adventure is a complex one, certainly the most cryptic in the book and one that should give you hours and hours of enjoyment. At the beginning of the game you will be asked to input a level. The level approximates to the number of moves allowed, and I suggest that initially you stick to a high number. There are a few red herrings but most of the game is an integral part of solving the problem, even down to your choice of clothes.



ADVENTURES FOR YOUR  
ZX SPECTRUM

```

9 REM ***EVERYDAY ADVENTURE***
10 INK 2
11 FOR t=0 TO 7: PAPER t: BEEP 0.2
8,t*t: CLS : NEXT t
12 FOR t=1 TO 52 STEP 4: CIRCLE 12
5,70,t: BEEP .025,t: NEXT t
13 INK 4
15 PRINT AT 2,7;"EVERYDAY ADVENTUR
E"
16 INK 3
35 PRINT ^0;TAB (10); FLASH 1;"Plse
ase Wait";
40 LET dk=0: LET ok=0: LET p=0: LE
T ww=0: LET sc=0: LET l=1: LET mv=0:
LET n=0
: LET mn=0: LET nw=0: LET cu=0: LET
nr=300: GO SUB 2170
50 REM ****Main Loop****
60 IF o(24)=-1 AND o(23)=-1 THEN
LET cf=3
70 IF o(13)=1 AND sc=0 THEN LET b
(1,1)=r(1,1): LET b(1,2)=r(1,2): LET
b(1,3)=
r(1,3): LET b(1,4)=r(1,4): LET r(1,1
)=0: LET r(1,2)=0: LET r(1,3)=0: LET
r(1,4)=
0: LET sc=1
80 IF nw=3 THEN GO TO 1920
90 PAUSE 0
100 IF nr=mv-10 AND cf<>1 THEN PRI
NT "You Are Wet Through"
110 LET mv=mv+1: IF mv>1v THEN LET
l=37
120 BORDER INT (RND*7): PAPER 0: CL

```

```

S : INK 7; BEEP .05,10: BEEP .07,0
  130 PRINT AT 0,7;"EVERYDAY ADVENTUR
E"
  140 PRINT AT 1,7;"=====
=": PRINT
  150 IF 1=37 THEN PRINT TAB (11); F
LASH 1;"Mums Home": PRINT
  160 PRINT "You Are...": PRINT r$(1)
  170 PRINT : PRINT "You Can See..."
  180 IF 1=37 THEN PRINT "Little Hap
piness For Yourself Over The Weeks
That Li
e Ahead": BEEP .2,10: BEEP .5,0: STO
P
  190 FOR t=1 TO 30: IF o(t)=1 THEN
PRINT o$(t): NEXT t
  195 NEXT t
  200 IF o(16)=1 AND (o(14)=0 OR o(18
)=0) THEN PRINT "The Policeman says
""Watch
It SoniHope You Don't Sniff That""
  210 IF o(13)=o(16) AND o(13)=1 THEN
PRINT "You're OK, The Policeman Ar
restsPet
e For Threatening Behavior": LET o(1
3)=12: LET o(16)=102
  220 IF 1=36 THEN PRINT "It Starts
Raining": LET rn=1: LET nr=mv: IF cf
=1 THEN
PRINT "It Does Not Bother You"
  230 IF o(16)=1 AND cf=1 THEN PRINT
"The Policeman Arrests You For We
aring Ho
bnail Boots Which Are Regarded As Of

```

```

offensive Weapons.": LET mv=mv+8
240 IF o(16)=1 THEN IF o(11)=0 THE
N PRINT "You Waste Much Time Explai
ning Th
at Your Umbrella Is Not An Offens
ive Weapon": LET mv=mv+8
250 IF ww=1 AND l=24 THEN PRINT "A
Bus Appears. Do You Wish To Aligh
t?": INP
UT w$: IF w$="yes" OR w$="y" THEN P
RINT "You're On The Bus": LET l=25:
LET mn=m
n-1: GO TO 260
255 IF ww=1 AND l=24 THEN PRINT b$
: LET mv=mv+INT (lv/2)
260 IF mn=0 AND o(1)=0 THEN PRINT
"You Have No Money Left": LET o(1)=1
02
270 IF l<14 AND o(26)=13 AND o(2)>3
0 AND o(6)>30 AND o(7)>30 AND o(8)>3
0 AND o(
9)=14 THEN GO TO 1970
280 PRINT : PRINT "What Do You Want
To Do?"
290 INPUT c$: IF c$="place vase" OR
c$="place new vase" OR c$="replace
vase" OR
c$="replace new vase" THEN LET o(2
6)=1: PRINT "Right you are": GO TO 5
0
300 IF LEN (c$)>2 THEN GO TO 400
310 IF (c$="n" OR c$="s") AND l=7 T
HEN PRINT "You Fall Off The Stairs
And Are

```

```

Unconscious Until Mum Comes Home": L
ET 1=37: PAUSE 0: NEXT t: GO TO 50
  320 IF 1=7 AND c$="u" THEN LET c$=
  "n"
  330 IF 1=7 AND c$="d" THEN LET c$=
  "s"
  340 IF c$="n" AND r(1,1)<>0 THEN L
ET 1=r(1,1): GO TO 390
  350 IF c$="s" AND r(1,2)<>0 THEN L
ET 1=r(1,2): GO TO 390
  360 IF c$="e" AND r(1,3)<>0 THEN L
ET 1=r(1,3): GO TO 390
  370 IF c$="w" AND r(1,4)<>0 THEN L
ET 1=r(1,4): GO TO 390
  380 PRINT "You Cannot Go That Way":
  BEEP .2,1: GO TO 50
  390 PRINT : PRINT "*****You're Mo
ving Now*****": PAUSE 0: NEXT t:
GO TO 50
  400 GO SUB 2090: PRINT : LET x=425
  401 IF n=0 OR n=1 THEN GO SUB 450:
GO TO x
  402 IF n=2 THEN GO SUB 540: GO TO
x
  403 IF n=4 OR n=5 THEN GO SUB 610:
GO TO x
  404 IF n=5 THEN GO SUB 690: GO TO
x
  405 IF n=6 OR n=7 THEN GO SUB 720:
GO TO x
  406 IF n=8 OR n=31 OR n=32 OR n=33
THEN GO SUB 780: GO TO x
  407 IF n=9 THEN GO SUB 960: GO TO
x

```

ADVENTURES FOR YOUR  
ZX SPECTRUM

```

408 IF n=10 THEN GO SUB 1050: GO T
O x
409 IF n=11 OR n=12 THEN GO SUB 11
10: GO TO x
410 IF n=13 THEN GO SUB 1180: GO T
O x
411 IF n=14 THEN GO SUB 1230: GO T
O x
412 IF n=15 THEN GO SUB 1300: GO T
O x
413 IF n=16 THEN GO SUB 1380: GO T
O x
414 IF n=17 THEN GO SUB 1500: GO T
O x
415 IF n=18 OR n=24 THEN GO SUB 15
50: GO TO x
416 IF n=19 OR n=20 OR n=21 THEN G
O SUB 1640: GO TO x
417 IF n=22 THEN GO SUB 1710: GO T
O x
418 IF n=23 THEN GO SUB 1760: GO T
O x
419 IF n=25 OR n=26 THEN GO SUB 18
10: GO TO x
420 IF n=27 THEN GO SUB 1880: GO T
O x
421 IF n=28 THEN GO SUB 1890: GO T
O x
422 IF n=29 THEN GO SUB 1900: GO T
O x
423 IF n=30 THEN GO SUB 1910
425 IF ok=1 THEN LET ok=0: PAUSE 0
: NEXT t: GO TO 50
430 PRINT : PRINT "*****OK

```



```

!*****"
440 GO TO 50
450 REM ***Take/Get***
460 IF p=6 THEN PRINT "You Are Car
rying As Much As You Can": LET ok=1:
RETURN
470 FOR t=1 TO 28: IF g$( TO 4)=z$(
t) AND o(dv)=1 THEN LET dv=t: LET t
=28: NEX
T t: GO TO 490
475 NEXT t
480 PRINT "There Is No Such Thing H
ere": RETURN
490 IF dv=2 OR dv=7 OR dv=12 OR dv=
13 OR dv=15 OR dv=16 OR dv=6 OR dv=2
8 THEN
PRINT "Don't Be Stupid, Of No Use":
LET dv=0: LET ok=1: RETURN
500 IF dv=10 THEN PRINT "It Is Glu
ed To The Pavement": RETURN
510 IF l=29 OR l=31 OR l=32 THEN P
RINT "You Have To Buy The Object In
Questi
on": RETURN
520 LET o(dv)=0: IF o(1)=0 AND l=1
THEN LET mn=3
525 BEEP .05,20
530 RETURN
540 REM ****Drop****
550 IF o(12)=1 AND g$="money" AND m
n>0 THEN LET mn=mn-1: PRINT "Oh Ta
Mate, He
h I'll Give You A Tip...Look Out For
The BeautifulFemale In The Cul De S

```

```

ac In Th
e Town Centre": FOR t=1 TO 500: NEXT
t: RETURN
560 IF o(26)=0 AND 1<>13 AND g$="ne
w vase" THEN PRINT "I Suggest You P
lace It
In The Original Room": RETURN
570 IF o(26)=0 AND 1=13 THEN PRINT
FLASH 1;"SMASH";: PRINT " You Don'
t Drop V
ases. ThatIs What Happened At The Pa
rty Last Night!": LET o(26)=31: LE
T o(8)=1
3: RETURN
580 IF o(dv)=0 THEN LET o(dv)=1: R
ETURN
590 IF dv>0 AND dv<99 THEN PRINT "
You're Not Carrying That Object": LE
T ok=1:
RETURN
600 PRINT "What Are You Talking Abo
ut?": LET ok=1: RETURN
610 REM ****Clean/Tidy****
620 LET kk=0
630 IF o(dv)=2 OR o(dv)=10 OR o(dv)
=12 OR o(dv)=13 THEN LET kk=1
640 IF 1=2 AND kk=1 THEN PRINT "Th
ey Are Put Away Tidily": LET o(2)=10
2: RETUR
N
650 IF kk=1 AND o(9)=0 AND 1<>o(dv)
THEN PRINT "You Cannot Clean Up So
meThing
That Is Not There!": RETURN

```

```

660 IF kk=1 AND l=o(dv) AND o(9)<>0
THEN PRINT "What With Your Hands?"
: RETURN

```

```

670 IF kk=1 AND l=o(dv) AND o(9)=0
THEN PRINT "The Mess Is Cleaned Up
Fast. Yo
uWould Make A Good Maid": LET o(dv)=
102: RETURN

```

```

680 PRINT "You Cannot Do That!": RE
TURN

```

```

690 REM ****Wear****

```

```

700 IF dv>22 AND dv<26 AND o(dv)=0
THEN PRINT "You Are Wearing The ";b
$(dv): L

```

```

ET o(dv)=-1: RETURN

```

```

720 REM ****Run/Flee****

```

```

730 IF o(13)<>1 AND o(16)<>1 THEN
PRINT "What Are You Running From?":
RETURN

```

```

740 IF (o(13)<>1 OR o(16)=1) AND cf
=2 THEN PRINT "You Get Away With Ea
se": LET

```

```

o(13)=1+INT (RND*6): LET o(16)=1-IN
T (RND*3): GO TO 770

```

```

750 IF INT (RND*5)=1 THEN PRINT "Y
ou Get Away But It Was Close": LET m
v=mv+3:

```

```

LET o(13)=1+INT (RND*3): LET o(16)=1
-INT (RND*3): GO TO 770

```

```

760 IF (o(13)=1 OR o(16)=1) AND cf<
>2 THEN PRINT "It Didn't Work, He C
aught Yo

```

```

u": LET mv=mv+5: RETURN

```

```

770 LET r(1,1)=b(1,1): LET r(1,2)=b
(1,2): LET r(1,3)=b(1,3): LET r(1,4)
=b(1,4):
LET sc=0: RETURN
780 REM ***Kill/Murder/Destroy*
790 CLS : PRINT
800 PRINT FLASH 1;"*****
*****";
810 PRINT FLASH 1;"*"; FLASH 0;TAB
(31); FLASH 1;"*";
820 PRINT FLASH 1;"*"; FLASH 0;"
Tut! Tut! You Seriously "; FLASH
1;"*";
830 PRINT FLASH 1;"*"; FLASH 0;"
Didn't Expect This game To "; FLASH
1;"*";
840 PRINT FLASH 1;"*"; FLASH 0;"
Have That Sort Of Feature, "; FLASH
1;"*";
850 PRINT FLASH 1;"*"; FLASH 0;"
Did You? Those Obvious, "; FLASH
1;"*";
860 PRINT FLASH 1;"*"; FLASH 0;"
Crude Commands Do Not Work "; FLASH
1;"*";
870 PRINT FLASH 1;"*"; FLASH 0;"
On A Game As Subtle And "; FLASH
1;"*";
880 PRINT FLASH 1;"*"; FLASH 0;"
Imaginative As This. You "; FLASH
1;"*";
890 PRINT FLASH 1;"*"; FLASH 0;"
Lose A Few Minutes Of Your "; FLASH
1;"*";

```

```

900 PRINT FLASH 1;"*"; FLASH 0;"
Time As Punishment.          "; FLASH
1;"*";
910 PRINT FLASH 1;"*"; FLASH 0;TAB
(31); FLASH 1;"*";
920 PRINT FLASH 1;"*"; FLASH 0;TAB
(31); FLASH 1;"*";
930 PRINT FLASH 1;"*****
*****";
940 FOR t=1 TO 3000: IF INKEY$<>""
THEN LET t=3000: NEXT t
945 NEXT t
950 RETURN
960 REM ****Hit****
970 IF dv<>27 AND dv<>12 AND dv<>13
AND dv<>16 AND o(dv)=1 THEN PRINT
"It Is N
ot Right To Take Your Temper Out
On Inanimate Objects": RETURN
980 IF dv=27 AND o(28)=1 AND b$(t)=
"girl" THEN PRINT "What A Way To Tr
eat A La
dy! YourBad Manners Has Lost You Hop
e OfSuccess": LET l=37: RETURN
990 IF dv=12 AND o(dv)=1 THEN PRIN
T "He Only Asked To Borrow Some M
oney. Th
at Was A Bit Cruel. He Skulks Away,
You May Not See HimFor Months": LET
o(12)=10
2: RETURN
1000 IF dv=13 AND o(dv)=1 THEN PRIN
T "You Must Be Made, This Guy Is T
wice The

```

Size Of You"

```

1010 IF dv=13 AND o(dv)=1 AND cf=1 T
HEN PRINT "But A Swift Kick With Yo
ur Boots
Sorted Him Out": LET o(13)=INT (RND*
36): LET r(1,1)=b(1,1): LET r(1,2)=b
(1,2): L
ET r(1,3)=b(1,3): LET r(1,4)=b(1,4):
LET sc=0: RETURN
1020 IF dv=13 AND o(dv)=1 THEN PRIN
T "You Wake Up Dazed. You Are Back I
n Your H
ouse. You Are Covered In Bruises An
d Cuts, Your Money Is Gone": LET mn=
0: RETUR
N
1030 IF dv=16 AND o(16)=1 THEN PRIN
T "What A Stupid Thing To Do! The P
oliceman
Replies With His Truceon. Ouc
h!": LET l=37: RETURN
1040 RETURN
1050 REM ****Dri****
1060 IF o(19)<>0 AND o(20)<>0 AND o(
21)<>0 AND o(22)<>0 THEN PRINT "You
Have To
Take It first": RETURN
1070 IF (dv=19 OR dv=21 OR dv=22) AN
D l=34 THEN PRINT "A Good, Strong D
rink Tha
t": LET dk=dk+1: LET o(dv)=102
1080 IF dk=3 THEN PRINT "You Stumbl
e All Over The Place": PRINT :,,,"Yo
u Are Se

```

```

eing Seeing Double! Double!": LET
l=37
1090 IF dv=20 AND l=34 THEN PRINT "
You're A Sensible Person, The Drin
k Refres
hes You Greatly."
1100 RETURN
1110 REM ****Help/Clue****
1120 LET v$=""
1130 IF INT (RND*7)=1 THEN LET v$="
The Town Is Where The Action Is": GO
TO 1170
1140 FOR t=1 TO 9: IF h(t)=1 THEN L
ET v$=h$(t): LET t=9: NEXT t
1145 NEXT t
1150 IF v$="" THEN LET v$="There's
No Help Here Bud!": GO TO 1170
1160 IF INT (RND*6)=1 THEN LET v$="
I'm Not Going To Help You Here"
1170 PRINT "Clue:": PRINT v$: RETURN

1180 REM ****Wait****
1190 IF l<24 AND l>26 THEN PRINT "T
ime Passes...": FOR t=1 TO 4000: NEX
T t: RET
URN
1200 IF l=24 THEN LET ww=1
1210 IF l>24 AND l<27 THEN PRINT "T
he Bus Travels Onwards": PAUSE 10: N
EXT t: L
ET l=l+1: RETURN
1220 RETURN
1230 REM ****Buy****
1240 IF mn<1 THEN PRINT "You Have N

```

```

o Money, You Cannot Buy Anything":
RETURN
1250 IF 1=22 OR 1=29 OR 1=31 OR 1=32
THEN GO TO 1270
1260 PRINT "Put Your Money Away, The
re is Nothing Here To Buy": RETURN

1270 IF 1>19 AND o(dv)=1 THEN PRINT
"You Now Own ": PRINT o$(dv): LET o
(dv)=0:
LET mn=mn-1: RETURN
1280 IF 1=19 AND o(12)=1 AND g$="and
y" THEN PRINT "Try Dropping The Mon
ey": RET
URN
1290 PRINT "Please Rephrase That": R
ETURN
1300 REM ****Say****
1310 IF g$="i need help" THEN PRINT
"You're Using The Wrong Command": R
ETURN
1320 IF (g$="you look nice" OR g$="i
think you're beautiful" OR g$="i lo
ve you"
OR g$="you are very pretty" OR g$="y
ou're wonderful") AND o(27)=1 AND cf
>2 AND c
u=1 THEN LET cu=2: GO TO 1370
1330 IF (g$="hello mate" OR g$="hi")
AND o(12)=1 THEN PRINT "Hi!": RETU
RN
1340 IF o(27)=1 AND cf<3 THEN PRINT
"No Matter What You Say, Your At
tire Rui

```



```

ns The Effect": RETURN
1350 IF o(27)=1 AND cf=3 THEN PRINT
  "Your Comments Have No Effect On Me
  , Try So
  mething Else": RETURN
1360 IF o(27)<>1 THEN PRINT "He Doe
  s Not Understand": RETURN
1370 PRINT "Thankyou, You're Nice Yo
  urself. Can I Help You In Some Way":
  RETURN
1380 REM ****Ask****
1390 LET l$="": IF g$(LEN g$ TO LEN
  g$)="?" THEN LET g$=g$( TO LEN g$-1
  )
1400 IF cu=0 AND g$="do you come her
  e often" AND o(27)=1 AND cf<>1 THEN
  LET l$=
  "Now And Then, Handsome": LET cu=1
1410 IF cu=0 AND o(27)=1 AND g$="do
  you come here often" AND cf=1 THEN
  PRINT "U
  gh! What Are You Wearing?": LET o(27
  )=INT (RND*36)
1420 IF g$="how are you" AND (o(27)=
  1 OR o(12)=1 OR o(16)=1) THEN LET l
  $="Oh No
  t Too Bad": IF o(12)=1 THEN LET l$=
  l$+", But I'm Short Of Cash"
1430 IF g$="can you lend me some mon
  ey" AND cu=2 AND o(27)=1 THEN LET l
  $="Of Co
  urse, Here's A Fiver": LET mn=mn+5
1440 IF l=34 AND o(12)<>1 AND o(13)<
  >1 AND o(16)<>1 AND o(27)<>1 AND dk=

```

```

0 THEN
LET l$="Have A Drink First, On The H
ouse"
1450 IF l=34 AND dk>0 AND (g$="how c
an i get home" OR g$="which way home
" OR g$=
"How can i get back to my house") TH
EN LET r(30,1)=15: LET l$="Go To Th
e Superm
arket And Head North"
1460 IF o(16)=1 AND (g$="which way h
ome" OR g$="how can i get home" OR g
$="how c
an i get back to my house") THEN LE
T l$="That's Easy My Lad, Go To The
Superm
arket And Head North Or GoEast At Th
e Post Box": LET r(30,1)=15
1470 IF (g$="what time is it" OR g$=
"what is the time") AND (o(16)=1 OR
l=35) TH
EN LET l$="It Is About 3.30 My Lad"
1480 IF l$="" THEN LET l$="Your Que
stion Has Errors"
1490 PRINT , ,l$: LET ok=1: RETURN
1500 REM ****Read****
1510 IF z$(dv)="note" AND o(dv)=0 T
HEN PRINT "It Reads: Will Be Back O
n The
4th At About 4.00. Love Mum xxx": PA
USE 0: RETURN
1520 IF z$(dv)="cale" AND l=6 THEN
PRINT "It Reads: Monday 4th April":
RETURN

```

```
1530 IF dv=15 AND o(15)=1 THEN PRINT "It Reads 'Spring Cleaning Sale- Great Bargains On All Cleaning Tools At The Hardware Shop, Towncentre'": RETURN
1540 PRINT "You Cannot Read That": RETURN
1550 REM ****Look/Examine****
1560 IF z$(dv)="girl" AND o(dv)=1 THEN PRINT "She Slaps You Round The Face And Walks Off. That Is No Way To Treat A Lady": FOR t=1 TO 100: PAUSE 0: NEXT t: LET ok=1: LET o(dv)=INT (RND*36): RETURN
1570 IF z$(dv)="coin" AND l=o(dv) THEN PRINT "Standard 10p Coin Glued To Pavement": LET ok=1: RETURN
1580 IF dv=4 OR dv=5 OR dv=15 THEN PRINT "Try A Different Command": RETURN
1590 IF (l>22 AND l<26) AND dv>22 AND dv<26 THEN PRINT "Good Quality And Worth The Price Too": RETURN
1600 IF dv=8 AND l=o(dv) THEN PRINT "It Looks In Bad Shape - 8 Pieces In All.": RETURN
1610 IF dv=28 AND l=8 AND mv<15 THEN PRINT "The Time Is 2.00 p.m.": RETURN
```

```

1620 IF dv=28 AND l=8 AND mv>15 THEN
PRINT "It Stopped At 3.10 p.m.": R
RETURN
1630 PRINT "You Cannot Examine That
Buddy": RETURN
1640 REM ***Glue/Stick/Sniff***
1650 IF a(14)<>0 AND a(18)<>0 THEN
PRINT "What With, Thin Air?"
1660 IF (a(14)=0 OR a(18)=0) AND z$(
t)="pete" THEN PRINT "Inventive But
It Won'
t Work"
1670 IF a(18)=0 AND z$(t)="poli" THE
N PRINT "Amazing It Worked, You're
Safe"
1680 IF a(8)=0 AND a(14)=0 THEN PRI
NT "Wrong Type Of Glue": RETURN
1690 IF a(8)=0 AND a(18)=0 AND INT (
RND*3)=1 THEN PRINT TAB (10);"It Wo
rked!!!"
: LET a(8)=102: LET a(26)=0: RETURN
1700 PRINT "No Good You Glued Yourse
lf! You Will Have To Wait Until Mum
GetsHome
To Fix It": LET l=37: FOR t=1 TO 10
00: NEXT t: RETURN
1710 REM ****Open****
1720 IF dv=11 AND a(dv)=0 THEN PRIN
T "Umbrella Up ";
1730 IF rn=0 THEN PRINT "But There
Is No Need": LET um=1: RETURN
1740 PRINT "That Will Keep You D
ry": LET rn=0: LET um=1: RETURN
1750 IF um=1 THEN LET p=p+1: IF p=6

```

```

THEN PRINT "You Were Carrying Too
Much, You
u Dropped Your Umbrella": LET o(11)=
1: LET um=0: RETURN
1760 REM ****Close****
1770 IF o(11)<>0 THEN PRINT "You Ca
n Only Open The Umbrella And You Do
Not Hav
e One"
1780 IF o(11)=0 AND um=0 THEN PRINT
"It Is Already Closed"
1790 IF o(11)=0 AND z$(t)="umbr" AND
um=1 THEN PRINT "It Is Closed";: I
F rn=1 T
HEN PRINT "It Is Raining Hard": LET
um=0
1800 PRINT : RETURN
1810 REM ****Inventory/List****
1820 LET p=1
1830 CLS : PRINT TAB (6); FLASH 1;"I
NVENTORY"
1840 PRINT TAB (6); "=====": PRIN
T : PRINT "You Are Carrying...": PRI
NT
1850 FOR t=1 TO 26: IF o(t)<1 THEN
PRINT " "; FLASH 1;p; FLASH 0;")";o$(
t): LET
p=p+1: NEXT t
1860 NEXT t: PAUSE 0
1870 RETURN
1880 PRINT "If I Got The Chance I Wo
uld!!!": LET ck=1: LET nw=nw+1: RETU
RN
1890 PRINT "Your French Is Most Appe

```

```

aling": LET ok=1: LET nw=nw+1: RETURN
N
1900 PRINT "You Are An Earthy Devil!
!!": LET ok=1: LET nw=nw+1: RETURN
1910 PRINT "That's Not Very Nice Now
Is It?": LET ok=1: LET nw=nw+1: RET
URN
1920 REM ****A*Telling*Off****
1930 CLS
1940 PRINT ,," You Had To Persist I
n Being      Rude. That Is Not The Wa
y To
Go Through Life. You Need To      Be T
aught A Lesson..."
1950 PRINT ,,"This Program Will Now
Disappear"
1960 IF nw=3 THEN FOR t=1 TO 500: N
EXT t: NEW
1970 REM ****A*Win*At*Last!****
1980 REM
1990 REM
2000 CLS
2010 PRINT ,,,," Your Mother Is Hom
e, She LooksAll Round The House And
Seeing
It Clean And Tidy Is Most"
2020 PRINT ,,"Impressed. So Impresse
d That SheGives You A Present. You T
ake  Of
f The Wrapper To Find..."
2025 FOR t=1 TO 500: NEXT t
2030 FOR i=7 TO 0 STEP -1: BORDER i:
BEEP .05,i: BEEP .05,-i: NEXT i: FO
R i=0 TO

```

```

7: BORDER i: BEEP .05,i: BEEP .05,-
i: NEXT i
2040 PRINT ,,,TAB (4): FLASH 1: INK
1:"A Vase Just Like Hers!"
2050 FOR t=1 TO 10
2060 BEEP .055,t: BEEP .15,-t
2070 NEXT t
2080 STOP
2090 REM ***Command*Handling***
2100 LET dv=0: LET t=0
2110 LET g$="": LET f#=c$( TO 3)
2120 FOR t=1 TO LEN c$: IF c$(t TO t
)= " " THEN LET g#=c$(t+1 TO ): LET
t=LEN (c
$): NEXT t
2125 NEXT t
2130 FOR t=1 TO 97 STEP 3: IF d$(t T
O t+2)=f$ THEN LET n=INT ((t+2)/3):
LET t=9
7: NEXT t: GO TO 2140
2135 NEXT t: PRINT ,,"Your Request:
":c$: PRINT " Is Not Possible": GO T
O 50
2140 IF g$="" THEN LET g$=" "": G
O TO 2150
2145 FOR t=1 TO 28: IF z$(t)=g$( TO
4) THEN LET dv=t: LET t=29: NEXT t
2146 NEXT t
2150 IF dv=0 THEN LET dv=30
2160 RETURN
2170 REM *Data/Initialisation**
2180 RESTORE
2190 DIM r$(37,87): DIM o$(30,35): D

```

```

1M o(30): DIM r(37,4): DIM z$(30,4):
  DIM h$(
9,20): DIM h(9): DIM b(37,4)
2200 FOR t=1 TO 37: READ r$(t): NEXT
  t
2210 FOR t=1 TO 30: READ o$(t): READ
  o(t): NEXT t
2220 FOR t=1 TO 30: READ z$(t): NEXT
  t
2230 FOR x=1 TO 37
2240 FOR y=1 TO 4: READ r(x,y): NEXT
  y: NEXT x
2250 FOR t=1 TO 9: READ h$(t): READ
  h(t): NEXT t
2260 DATA "In Your Bedroom","In Your
  Sisters Bedroom","In Your Parents B
  edroom",
  "In The Bathroom","On The Landing","
  On The Landing","On The Stairs"
2270 DATA "In The Hall","In The Hall
  ","In The Kitchen","Within The Study
  ","In Th
  e Dining Room","Inside The Sitting R
  oom"
2280 DATA "Out On The Patio","In The
  Middle Of Your Garden","Out In Your
  Street"
  ,"In The Street","In The Middle Of T
  he Local Park"
2290 DATA "In A Small Newsagents","N
  ext To A Post Box","Down A Small Lan
  e","By T
  he Main Road","Further Along The Mai
  n Road","By A Bus Stop","In The Bus

```



Next To

Another Stop"

2300 DATA "In The Bus Next To A Bus Stop", "By The Last Stop On The Route , The B us Has Driven Off, You Are In The To wn Centre", "Down A Cul De Sac", "Brow sing In A Fashion Store"

2310 DATA "Inside A Supermarket", "Wi thin A Tiny Pottery Shop", "In The Ha rdware S tore", "In The Car Park", "In The Red Lion", "Inside The Police Station", "I n The Mi ddle Of The High Street", "In The Dog House, My Friend!!!"

2320 DATA "Money", 1, "Records All Ove r The Floor", 2, "A Rubber Duck", 4, "A Calender ", 6, "A Note For You", 9, "A Mass Of Bo ttles", 10, "Crisps & Peanuts Over The Floor", 12, "Mum's Broken Vase", 13, "Some Clea ning Tools", 14

2330 DATA "A 10p Coin", 16, "An Umbrel la", 23, "Your Pal Andy", 22, "Bully, Pe te", 21, " A Tube Of Blue", 20, "A Notice Board", 19, "A Policeman", 35

2340 DATA "A Broom", 32, "A Tube Of Wo nder Blue", 32, "A Wine Box", 34, "An Or ange Jui ce", 34, "A Gin And Tonic", 34, "A Bottl

```

e Of Vodka",34,"A New Jacket",29
2350 DATA "A Pair Of Shoes",29,"A St
ring Vest",29,"A New Vase",31,"A Pre
tty Girl
",28,"A Grandfather Clock",8,"",102,
"Nothing",102
2360 DATA "mone","reco","duck","cale
","note","bott","cris","brok","clea
","coin",
"umbr","andy","pete","glue","noti","
poli"
2370 DATA "broc","wond","wine","oran
","gin ","vodk","jack","shoe","stri
","new ",
"girl","cloc","","noth"
2380 DATA 2,5,0,0,0,1,0,0,0,6,4,0,0,
0,0,3
2390 DATA 1,0,6,0,3,7,0,5,6,8,0,0,7,
16,9,10
2400 DATA 0,0,13,8,12,0,8,11,12,0,10
,0,0,11,10,13
2410 DATA 14,9,12,0,15,13,0,0,20,14,
0,0,0,0,17,8
2420 DATA 22,18,0,16,17,19,20,0,18,0
,0,0,21,0,15,18,23,20,0,0,0,17,23,0,
0,21,24,
22
2430 DATA 0,0,0,23,0,0,0,0,0,0,0,2
8,36,0,0
2440 DATA 0,27,0,0,0,0,30,0,30,31,0,
29,30,32,0,34
2450 DATA 31,0,0,33,34,0,32,35,0,33,
31,36,36,0,33,0,27,35,34,0,0,0,0,0
2460 LET d$="takdrocletidwearunfleki

```

```

1hitdrihelcluwaibuysayaskreaexastifi
xglucopec
1000invlisfuccunsodarsmurdesanh"
2470 LET b$="The Bus Roars Off, It W
as The Only Way Into The Town. You
HaveTo
Wait An Hour For The Next One"
2480 DATA "Bathtime Is Fun",4,"Take
The Note First",9,"South To The Outs
ide",8,"
A Pause Would Be Advisable",24,"Rela
x And Wait",25,"Remember Your Manner
s And Da
tingCliches",28,"Don't Beat The Mast
er At His OwnGame",21
2490 DATA "Ask For Advice On Getting
Home",34,"Ask For Advice On Getting
Home",3
5
2500 REM **Choose*Your*Status**
2510 CLS
2520 PRINT : PRINT " Just One Deci
sion To Be Made Before You Proceed."
: PRINT
2530 INPUT "Your Choice Of Level (2-8
)":lv
2535 PRINT ^0;"
";
2540 IF lv<2 OR lv>8 THEN GO TO 253
0
2550 LET lv=lv*20
2560 PAUSE 0
2570 CLS
2580 PRINT ,,,," You Wake Up On M

```

Monday, You Have No Idea Of The Time  
"

2590 PRINT "., " You Need To Get Dr  
essed, You Do So In Quite A Hurry...W  
hy? Yo

u're Not Sure But You Know Someth  
ing Happened Last Night..."

```
2600 FOR t=1 TO 500: IF INKEY$("<>")="" T
HEN LET t=500: NEXT t
2605 NEXT t
```



```
2610 PRINT ,," You Have Got Dressed  
Except YouAre Not Wearing Shoes Or A  
Coat (I  
t's Freezing In Your House)"  
2620 PRINT ,," You Have The Followin  
g Choices:"  
2630 PRINT FLASH 1;"1"; FLASH 0;" )  
A Souwester And Hobnail Boots";  
2640 PRINT FLASH 1;"2"; FLASH 0;" )  
A Tracksuit And Trainers"  
2650 PRINT FLASH 1;"3"; FLASH 0;" )  
A Suede Coat And Your Best Shoe  
s"  
2660 INPUT "Enter Your Choice(1,2,3)  
?";cf: IF cf<1 OR cf>3 THEN GO TO 2  
660  
2670 PRINT ^0;"  
";  
2680 PRINT ,,,,"Right. Now You Can S  
tart": FOR t=1 TO 30 STEP .5: BEEP .  
0015,t:  
NEXT t: RETURN
```

# WRITING YOUR OWN ADVENTURES

If you really enjoy playing adventures then there is a good chance that you will enjoy writing your own adventures, based on your own plot, even more. Actually constructing your own adventure is not as difficult as it sounds and once you have written a couple of adventures you will appreciate how much fun it can be.

Firstly, let me distinguish the adventure game from other games and simulations. In my opinion, the adventure game has the following elements:

- 1) Locations.
- 2) Usually some form of logical connection between the locations (though the logic is sometimes far from perfect).
- 3) Some form of reward or aim to work for; escape, for instance, or the accumulation of treasure. Without some objective, an adventure would become pointless and boring.
- 4) Some form of obstacle hindering the progress of the adventurer.
- 5) Some element of discovery — find the wizard, the exit, how to make your character move, and so on.
- 6) Some form of command with which adventurers can direct themselves around the locations and thus interact with the adventure itself.

Before you write any of the actual program you should plan out a number of things. You must have a plot, and descriptions of the locations, characters, objects and the links between them all. In addition, I recommend that you work out all the obstructions, puzzles and connections between locations. This information will be a great help when you are writing and, more importantly, debugging your adventure.

Let's start with the plot of your adventure. This will involve a scenario and some form of objective(s), as well as some major hinderances. Plots can be easy to come by,

searching for gold, finding your way out of a maze, rescuing a trapped maiden. Thinking up a novel plot is much more difficult but it does make for a vastly improved game. It can get rather tiring sitting in front of a screen trying to accumulate more treasure from a similarly-named castle that is guarded by similar-type monsters. What your adventure needs is something new or at least a little less common. 'Crash!', the adventure set inside the computer, is an example of such a plot.

The actual objectives and the scenario, or playing environment, are closely connected. Once you have determined one, the other is simple to think of. If you choose an underwater scenario, then an obvious objective would be to find a sunken wreck. Similarly, if you choose escaping from a maze as your objective then a mythological scenario with gods and minotaur-type monsters immediately springs to mind. Below are a few other suggested scenarios and objectives:

- |                             |   |
|-----------------------------|---|
| <b>Small Island</b>         | You have been shipwrecked and must find a way to get off the island.  |
| <b>Office Building</b>      | As a secret agent, you are searching for some top secret files.   |
| <b>The Human Body</b>       | As an antibody you must find the source of infection and fight it fast.   |
| <b>Mysterious House</b>     | You have gone on a mystery tour for some fun, things have turned dangerous and you and several others have been left here. What are the secrets in the house? |
| <b>Houses of Parliament</b> | Can you find the leaked government report before the media do? Your chance of a place in the Cabinet depends on your success in this mission.                 |

### The London Underground

A plague of rats that have left the sewers are resting in the Underground system. Can you eradicate them before they are strong enough to attack again?

When you put your mind to it, the possibilities are limited only by your imagination. Why not set an adventure in your own house, where the main aim is to avoid doing the washing-up. Finding a good plot should not be that difficult and it really can help make your adventure an interesting one.

Once you have the plot, it is not too hard thinking up names and descriptions of locations. Obviously, some will need to be very specific as they tie in closely with the plot; in the human body scenario, all the locations will obviously have to be parts of the body. In most adventures, however, there is also a need for some less specific, even though not so interesting, locations. These locations add to the realism of the adventure; for instance, in the Houses of Parliament scenario, going from the main chamber in the House of Commons straight into the bar or the library may lack realism, and after a few locations, you will start to run out of names. Locations such as corridors, small storerooms, a dusty alcove, a windswept quadrangle and a committee room may add to the adventure.

The descriptions of the locations is a matter of personal taste. I do not like long-winded descriptions; memory space is still vital enough to be conserved for a greater number of locations or more fiendish puzzles. However, I know people who revel in the task of creating beautiful passages of descriptive prose for each location. If you decide to write long descriptions, it may be a good idea to have a shortened version of the location description which can be printed up once the player has seen the longer version. This makes the adventure more interesting to the user who can always read the long description again, accessing it by using the command, LOOK.



I will discuss characters in the same breath as objects since in adventure programming they are often treated in the same way. Your plot will obviously determine a number of the characters and the objects, but you may decide to add others either to create further obstructions or to aid the adventurer in his struggle to reach his goal. Although I can think of adventures where there are no characters, all adventures need objects of some kind. The objects and the characters are linked most strongly with the puzzles in the adventure, and this is the next part under discussion.

In an adventure, the plot and the puzzles are the two most important factors to be considered. Again, some of the puzzles will obviously come from the plot, but almost certainly you will need a number of extra puzzles to complete the adventure. One can categorise puzzles into three types. The first type of puzzle is directly linked to the plot and must be solved before the adventure can be finished. The second type of puzzle is not directly linked to the plot but still needs to be solved for the adventure to be completed. The final type of puzzle need not be solved; but while it may add to the realism and the fun of the adventure, it may also slow down the adventurer who is not aware that it is unimportant. Below are examples of the first two types of puzzle:

### **Need To Be Solved**

- 1) The castle is dark without the torch, you can still move around but each location is not described.
- 2) You need a gun if you want to get past the guard at the door. You must haggle with the trader in the village.
- 3) You have broken a vase in the office and, to get out without it being noticed, you must find some glue.
- 4) The door is not locked but it is too heavy for you to push open. It is huge and wooden.
- 5) You come to a large canyon, you cannot climb down and behind you is a herd of stampeding cattle. All you have is unlimited amounts of string. What are you going to do?

Those of you who enjoy lateral thinking puzzles should be able to solve the last problem. The solution is to fill the canyon with string, walk over the canyon — which is now full — and then quickly take out all the string.

You only need a couple of very difficult problems in an adventure but these should be supplemented with a number of 'red herrings'. Some suggested ones are indicated below:

- 1) You are carrying a torch, and you spot a toy chemical set; if the player examines or removes the set then the torch ignites it and the adventurer is blown to smithereens.
- 2) Is the rubber duck of any use? After playing whacky adventure games like Pimania no one can be sure.
- 3) A crimson fish (this one is too obvious for words!)

There are many quite simple puzzles which can be implemented in your own adventure: locked doors, creatures barring your way, I'm sure you know the sorts of puzzles.

You will need to make a 'map' of your adventure showing all the links between locations, positions of objects and so on. The majority of adventures use the compass points, N, S, E and W, for direction of movement, with maybe Up and Down. Your adventure can use these, as well as the four points — NE, NW, SE, SW — or it can use a different system. You will also need to formulate a program routine that transfers your position from one location to another. There are a number of different methods, from simple grid systems to complex systems that allow movement in 16 different directions (including through time!).

Once you have all this neatly down on paper, you can start the actual programming. As with all games, there are certain points to note. Make sure the screen is presented neatly with no words cut in two by the end of a line. Add a bit of sound and some graphics here and there but try to make them relevant to the program; that is, when someone fires a gun, have a sound indicating this, and when the adventurer has just killed a monster, turn the screen red. Small touches like these increase an

adventure game's interest. Graphic pictures of locations are fine as long as they appear quickly; there is nothing more tiresome than having to wait for ages while a screen fills with a drawing. The user will grow tired of the pictures very rapidly despite their quality.

Finally, a word on structuring your adventure program. Why bother to structure programs at all? There are a number of good reasons for doing so. Since BASIC is a poorly structured language — compared to a language such as FORTH — the programmer must add his own framework. Structuring a program speeds a program up, as it does not have to pass through irrelevant masses of code every loop. Structuring also eradicates duplication of program functions and helps to save much memory space that can be put to good use elsewhere.

Another reason, particularly applicable to adventure programmers, is that after leaving a program for a few days, you will start to forget where each routine is. By structuring a program, the adventure writer knows where each routine is and what it does, and this makes debugging a great deal easier.

After all this, I hope you will see that there is a lot of work to be done when writing an adventure. It is, however, within the bounds of most computer users. It is an ideal way to start programming once the beginner has written a few simple programs. Adventures, in short, whether you write them or play them, can be enormously enjoyable.

# GLOSSARY

## A

- Accumulator** — the place within the computer in which arithmetic computations are performed and where the results of these computations are stored.
- Algorithm** — the series of steps the computer follows to solve a particular problem.
- Alphanumeric** — this term is usually used in relation to a keyboard, as in 'it is an alphanumeric keyboard', which means that the keyboard has letters as well as numbers. It is also used to refer to the 'character set' of the computer. The character set comprises the numbers and letters the computer can print on the screen.
- ALU (Arithmetic/Logic Unit)** — the part of the computer which does arithmetic (such as addition, subtraction) and where decisions are made.
- AND** — a Boolean logic operation that the computer uses in its decision-making process. It is based on Boolean algebra, a system developed by mathematician George Boole (1815-64). In Boolean algebra the variables of an expression represent a logical operation such as OR and NOR.
- ASCII** — stands for American Standard Code for Information Exchange, the most widely used encoding system for English language alphanumerics. There are 128 upper and lower case letters, digits and some special characters. ASCII converts the symbols and control instructions into seven-bit binary combinations.
- Assembler** — a program which converts other programs written in assembly language into machine code (which the computer can understand directly). Assembly language is a low level programming language which uses easily memorised combinations of two or three letters to represent a particular instruction which the assembler then converts so the machine can understand it. Examples of these are ADD (add), and SUB (subtract). A computer programmed in assembly language tends to work more quickly than one programmed in a higher level language such as BASIC.

**B**

- BASIC** — an acronym for Beginners All-Purpose Symbolic Instruction Code. It is the most widely used computer language in the microcomputer field. Although it has been criticised by many people, it has the virtue of being very easy to learn. A great number of BASIC statements resemble ordinary English.
- Baud** — named after Baudot, a pioneer of telegraphic communications. Baud measures the rate of transfer of information and is approximately equal to one bit per second.
- BCD** — an abbreviation for Binary Coded Decimal.
- Benchmark** — a test against which certain functions of the computer can be measured. There are a number of so-called 'standard Benchmark tests', but generally these only test speed. This is rarely the aspect of a microcomputer that is most of interest to the potential buyer.
- Binary** — a numbering system that uses only zeros and ones.
- Bit** — an abbreviation for Binary Digit. This is the smallest unit of information a computer circuit can recognise.
- Boolean Algebra** — the system of algebra developed by mathematician George Boole which uses algebraic notation to express logical relationships (see AND).
- Bootstrap** — a short program or routine which is read into the computer when it is first turned on. It orients the computer to accept the longer, following program.
- Bug** — an error in a computer program which stops the program from running properly. Although it is generally used to mean only a fault or an error in a program, the term bug can also be used for a fault in the computer hardware.
- Bus** — a number of conductors used for transmitting signals such as data instructions, or power in and out of a computer.
- Byte** — a group of binary digits which make up a computer word. Eight is the most usual number of bits in a byte.

**C**

- CAI** — Computer Assisted Instruction.
- CAL** — Computer Assisted Learning. The term is

generally used to describe programs which involve the learner with the learning process.

**Chip** — the general term for the entire circuit which is etched onto a small piece of silicon. The chip is, of course, at the heart of the microcomputer.

**Clock** — the timing device within the computer that synchronises its operations.

**COBOL** — a high level language derived from the words Common Business Orientated Language. COBOL is designed primarily for filing and record-keeping.

**Comparator** — a device which compares two things and produces a signal related to the difference between the two.

**Compiler** — a computer program that converts high level programming language into binary machine code so the computer can handle it.

**Complement** — a number which is derived from another according to specified rules.

**Computer** — a device with three main abilities or functions:

- 1) to accept data
- 2) to solve problems
- 3) to supply results

**CPU** — stands for Central Processing Unit. This is the heart of the computer's intelligence, where data is handled and instructions are carried out.

**Cursor** — a character which appears on the TV screen when the computer is operating. It shows where the next character will be printed. On a computer there are usually 'cursor control keys' to allow the user to move the cursor around the screen.

## D

**Data** — information in a form which the computer can process.

**Debug** — the general term for going through a program and correcting any errors in it, that is, chasing down and removing bugs (see Bug).

**Digital Computer** — a computer which operates on information which is in a discrete form.

**Disk/Disc** — this is a magnetically sensitised plastic disk, a little smaller than a single play record. This is used for

storing programs and for obtaining data. Disks are considerably faster to load than a cassette of the same length program. The disk can be searched very quickly while a program is running for additional data.

**Display** — the visual output of the computer, generally on a TV or monitor screen.

**Dot Matrix Printer** — a printer which prints either the listing of a program or that which is displayed on the TV screen. Each letter and character is made up of a number of dots. The higher the number of dots per character the finer the resolution of the printer.

**Dynamic Memory** — a memory unit within the computer which 'forgets' its contents when the power is turned off.

## E

**Editor** — this term is generally used for the routine within the computer which allows you to change lines of a program while you are writing it.

**EPROM** — stands for Erasable Programmable Read-Only Memory. This is like the ROM in the computer, except that it is fairly easy to load material into an EPROM and it doesn't disappear when you turn the power off. EPROMs must be placed in a strong ultra violet light to erase them.

**Error Messages** — the information given by a computer where there is a fault in the coding during a part of a program, usually shown by the computer stopping, and printing a word, or a word and numbers, or a combination of numbers only, at the bottom of the screen. This tells you what mistake has been made. Common mistakes include using the letter O instead of zero in a line, or leaving out a pair of brackets, or one of the brackets, in an expression, or failing to define a variable.

## F

**File** — a collection of related items of information organised in a systematic way.

**Floppy Disk** — a relatively cheap form of magnetic disk used for storing computer information, and so named because it is quite flexible (see Disk/Disc).

**Flow Chart** — a diagram drawn up before writing a program, in which the main operations are enclosed within

rectangles or other shapes and connected by lines, with arrows to represent loops, and decisions written at the branches. It makes writing a program much easier because traps such as infinite loops, or non-defined variables can be caught at an early stage. It may not be worth writing a flow chart for very short programs, but generally a flow chart aids in creating programs.

**Firmware** — there are three kinds of 'ware' in computers: software 'temporary' programs; hardware like the ROM which contains permanent information; and firmware in which the information is relatively permanent, as in an EPROM (see EPROM).

**Flip-Flop** — a circuit which maintains one electrical condition until changed to the opposite condition by an input signal.

**FORTRAN** — an acronym for FORMula TRANslation, this is a high level, problem orientated computer language for scientific and mathematical use.

## G

**Gate** — an electrical circuit which, although it may accept one or more incoming signals, only sends out a single signal.

**Graphics** — pictorial information as opposed to letters and numbers.

## H

**Hard Copy** — computer output which is in permanent form.

**Hardware** — the physical parts of the computer (also see software and firmware).

**Hexadecimal (Hex)** — a numbering system to the base sixteen. The digits zero to nine are used, as well as the letters A, B, C, D, E and F to represent numbers. A equals 10, B equals 11, C equals 12, and so on. Hex is often used by microprocessor users.

**Hex Pad** — a keyboard designed specifically for entering hexadecimal notation.

**High Level Language** — a programming language which allows the user to talk to the computer more or less in English. In general, the higher the level of the language (that is, the



closer it is to English), the longer it takes for the computer to translate it into a language it can use. Lower level languages are far more difficult for human operators but are generally executed far more quickly.

## I

**Input** — the information fed into the computer via a keyboard, a microphone, a cassette or a disk.

**Input/Output (I/O Device)** — a device which accepts information or instructions from the outside world, relays it to the computer, and then, after processing, sends the information out in a form suitable for storing, or in a form which could be understood by a human being.

**Instruction** — data which directs a single step in the processing of information by the computer (also known as a command).

**Integrated Circuit** — a complete electronic circuit imprinted on a semiconductor surface.

**Interface** — the boundary between the computer and a peripheral such as a printer.

**Interpreter** — a program which translates the high level language fed in by the human operator, into a language which the machine can understand.

**Inverter** — a logic gate that changes the signal being fed in, to the opposite one.

**Interactive Routine** — part of a program which is repeated over and over again until a specified condition is reached.

## J

**Jump Instruction** — an instruction which tells the computer to go to another part of the program, when the destination of this move depends on the result of a calculation just performed.

## K

**K** — this relates to the size of the memory. Memory is usually measured in 4K blocks. 1K contains 1,024 bytes.

**Keyword** — the trigger word in a line of programming, usually the first word after the line number. Keywords include STOP, PRINT and GOTO.

## L

**Language** — computer languages are divided into three sections: high level languages, such as BASIC, which are reasonably close to English and fairly easy for humans to use; low level languages, such as Assembler, that use short phrases which have some connection with English (ADD for add and RET for return, for instance); and machine code which communicates more or less directly with the machine.

**LCD** — this stands for Liquid Crystal Diode. Some computers such as the TRS-80 Pocket Computer use an LCD display.

**LED** — this stands for Light Emitting Diode. The bright red numbers which are often used on watch or clock displays are made up of LEDs.

**Logic** — the mathematical form of a study of relationships between events.

**Loop** — a sequence of instructions within a program which is performed over and over again until a particular condition is satisfied.

## M

**Machine Language or Machine Code** — an operation code which can be understood and acted upon directly by the computer.

**Magnetic Disk** — see Disk and Floppy Disk.

**Mainframe** — computers are generally divided into three groups, and the group a computer falls into depends more or less on its size. The computer you are thinking of buying is a microcomputer; medium sized computers are known as minicomputers; and the giant computers that you sometimes see in science fiction movies are mainframe computers. Until 15 years ago mainframe computers were, in practical terms, the only ones available.

**Memory** — there are two types of memory within a computer. The first is called ROM (read-only memory); this is the memory that comes already programmed on the

computer, which tells the computer how to make decisions and how to carry out arithmetic operations. This memory is unaffected when you turn the computer off. The second type is RAM (random access memory). This memory holds the program you type in at the keyboard or send in via a cassette or disk. In most computers the computer 'forgets' what is in RAM when you turn the power off.

**Microprocessor** — the heart of any computer. It requires peripheral unit interfaces, such as a power supply and input and output devices, to act as a microcomputer.

**MODEM** — stands for Modulator Demodulator. This is a device which allows two computers to talk to each other over the telephone. The computers usually use a cradle in which a telephone receiver is placed.

**Monitor** — this has two meanings in computer terms. One meaning is a television-like display. A monitor has no facility for tuning television programs, and usually the picture produced on a monitor is superior to that produced by an ordinary television. The second meaning of a monitor relates to ROM. The monitor of a computer is described as the information it has built in when you buy it. This information allows it to make decisions and carry out arithmetic computations.

**Motherboard** — a framework to which extra circuits can be added. These extra circuits often give the computer facilities which are not built-in, such as that of producing sound or of controlling a light pen.

**MPU** — an abbreviation for Microprocessor Unit.

## N

**Nano-second** — a nano-second is one thousand billionth of a second, the unit of speed in which a computer or a memory chip is often rated.

**Non-Volatile Memory** — memory which is not lost when the computer is turned off. Some of the smaller computers such as the TRS-80 Pocket Computer have non-volatile memory. The batteries hold the program you enter for several hundred hours.

**Not** — a Boolean logic operation that changes a binary digit into its opposite.

**Null String** — a string which contains no characters. It is shown in the program as two double quote marks, without anything between them.

**Numeric** — pertaining to numbers as opposed to letters (that is, alphabetic). Many keyboards are described as being alphanumeric which means both numbers and letters are provided.

## O

**Octal** — a numbering system which uses eight as the base, and the digits 0, 1, 2, 3, 4, 5, 6 and 7. The Octal system is not used very much nowadays in microcomputer fields. The Hexadecimal system is more common (see Hexadecimal).

**Operating System** — the software or firmware generally provided with the machine that allows you to run other programs.

**OR** — an arithmetic operation that returns a 1, if one or more inputs are 1.

**Oracle** — a method of sending text messages with a broadcast television signal. A teletext set is required to decode the messages. Oracle is run by Independent Television Service in the UK, and a similar service — Ceefax — is provided by the BBC.

**Output** — information or data fed out by the computer to such devices as a TV-like screen, a printer or a cassette tape. The output usually consists of the information which the computer has produced as a result of running a program.

**Overflow** — a number too large or too small for the computer to handle.

## P

**Pad** — see Keypad.

**Page** — often used to refer to the amount of information needed to fill one TV screen, so you can talk about seeing a page of a program, the amount of the listing that will appear on the screen at one time.

**PASCAL** — a high level language.

**Peripheral** — anything which is hooked onto a computer, for control by the computer, such as a disk unit, a printer or a voice synthesiser.

**Port** — a socket through which information can be fed out of or in to a computer.

**Prestel** — the British telecom name for a system of calling up pages of information from a central computer via the

telephone and displaying them on a television screen. A similar commercial version in the United States is known as The Source.

**Program** — in computer terms program has two meanings. One is the list of instructions that you feed into a computer, and the second is used as a verb, as in 'to program a computer'.

**PROM** — stands for Programmable Read Only Memory. This is a device which can be programmed, and once it is then the program is permanent (also see EPROM and ROM).

## R

**Random Access Memory (RAM)** — the memory within a computer which can be changed at will by the person using the computer. The contents of RAM are usually lost when a computer is turned off. RAM is the memory device that stores the program that you type in and also stores the results of calculations in progress.

**Read-Only Memory (ROM)** — in contrast to RAM, information in ROM cannot be changed by the user of the computer, and the information is not lost when the computer is turned off. The data in ROM is put there by the manufacturers and tells the computer how to make decisions and how to carry out arithmetic computations. The size of ROM and RAM is given in the unit K (see K).

**Recursion** — the continuous repetition of a part of the program.

**Register** — a specific place in the memory where one or more computer words are stored during operations.

**Reserved Word** — a word that you cannot use for a variable in a program because the computer will read it as something else. An example is the word TO. Because TO has a specific computer meaning, most computers will reject it as a name for a variable. The same goes for words like FOR, GOTO and STOP.

**Routine** — this word can be used as a synonym for program, or can refer to a specific section within a program (also see Subroutine).

## S

**Second Generation** — this has two meanings. The first applies to computers using transistors, as opposed to first

generation computers which used valves. Second generation can also mean the second copy of a particular program; subsequent generations are degraded by more and more noise.

**Semiconductor** — a material that is usually an electrical insulator but under specific conditions can become a conductor.

**Serial** — information which is stored or sent in a sequence, one bit at a time.

**Signal** — an electrical pulse which is a conveyor of data.

**Silicon Valley** — the popular name given to an area in California where many semiconductor manufacturers are located.

**SNOBOL** — a high level language.

**Software** — the program which is entered into the computer by a user which tells the computer what to do.

**Software Compatible** — this refers to two different computers which can accept programs written for the other.

**Static Memory** — a non-volatile memory device which retains information so long as the power is turned on, but does not require additional boosts of power to keep the memory in place.

**Subroutine** — part of a program which is often accessed many times during the execution of the main program. A subroutine ends with an instruction to go back to the line after the one which sent it to the subroutine.

## T

**Teletext** — information transmitted in the top section of a broadcast television picture. It requires a special set to decode it to fill the screen with text information. The BBC service is known as Ceefax, the ITV service as Oracle. Teletext messages can also be transmitted by cable, for example the Prestel service in Britain or The Source in the United States.

**Teletype** — a device like a typewriter which can send information and also receive and print it.

**Terminal** — a unit independent of the central processing unit. It generally consists of a keyboard and a cathode ray display.

**Time Sharing** — a process by which a number of users may have access to a large computer which switches rapidly

from one user to another in sequence, so each user is under the impression that he or she is the sole user of the computer at that time.

**Truth Table** — a mathematical table which lists all the possible results of a Boolean logic operation, showing the results you get from various combinations of inputs.

## U

**UHF** — Ultra High Frequency (300-3000 megaHertz).

**Ultra Violet Erasing** — Ultra violet light must be used to erase EPROMs (see EPROM).

## V

**Variable** — a letter or combination of letters and symbols which the computer can assign to a value or a word during the run of a program.

**VDU** — an abbreviation for Visual Display Unit.

**Volatile** — refers to memory which 'forgets' its contents when the power is turned off.

## W

**Word** — a group of characters, or a series of binary digits, which represent a unit of information and occupy a single storage location. The computer processes a word as a single instruction.

**Word-Processor** — a highly intelligent typewriter which allows the typist to manipulate text, to move it around, to justify margins and to shift whole paragraphs if necessary on a screen before outputting the information onto a printer. Word-processors usually have memories, so that standard letters and the text of letters, written earlier, can be stored.

# BIBLIOGRAPHY

This area of computing has not been overflooded with publications, and only recently have a number of books come onto the market — some excellent, some awful. Below I mention only the good ones.

While a number of these books may not have been written specifically for your computer, the ones mentioned either have an informative text or adventures that can be easily converted to your machine.

## **Creating Adventure Games on Your BBC Micro**

Ian Watt. Interface/Addison Wesley

Ian really does know his stuff when it comes to writing adventure programs. He has his own style of adventure writing and, in this book, he reveals all. It is a slim volume containing three adventures all in text so they can be converted to other computers.

## **Creating Adventure Programs on Your Computer**

Andrew Nelson. Interface

I have met Andrew and he is full of interesting ideas, which he kindly shared with me. This book contains a number of adventures, all written in Microsoft BASIC — and easily converted to another computer. A particularly intriguing title is 'The Aftermath of the Asmorian Disaster'.

## **Adventure Writing**

Aardvark-80, 2352 S. Commerce, Walled Lake, MI 48088, USA

This 16-page booklet, sold in the United States (for the exorbitant sum of \$5), is a terrific help to all adventure writers. The adventure program included, 'Death Ship', is broken down in detail and comes with an addendum offering versions for most home computers.



**Creating Adventure Games on Your Dragon 32**

Clive Gifford. Interface

Five full adventures are explained in detail, one of which has now been transferred to cassette and disc software. Three of the five adventures can be converted for use on other computers without much difficulty.

**The ZX81 Pocket Book**

Trevor Toms. Phipps Associates

Only one section is devoted to adventures but, in it, the author details a vastly different approach to adventure writing. The book may be worth buying, particularly if you can find it at a discounted price.

**Creating Adventures on Your Spectrum**

Peter Shaw/James Mortleman. Interface

I know Peter well — he is on the editorial board of *Your Spectrum* magazine — and this is a strong book with many novel adventures, some featuring excellent graphics. Another point of note: the illustrations in this book were drawn by Peter himself.

**Writing BASIC Adventure Games for the TRS-80**

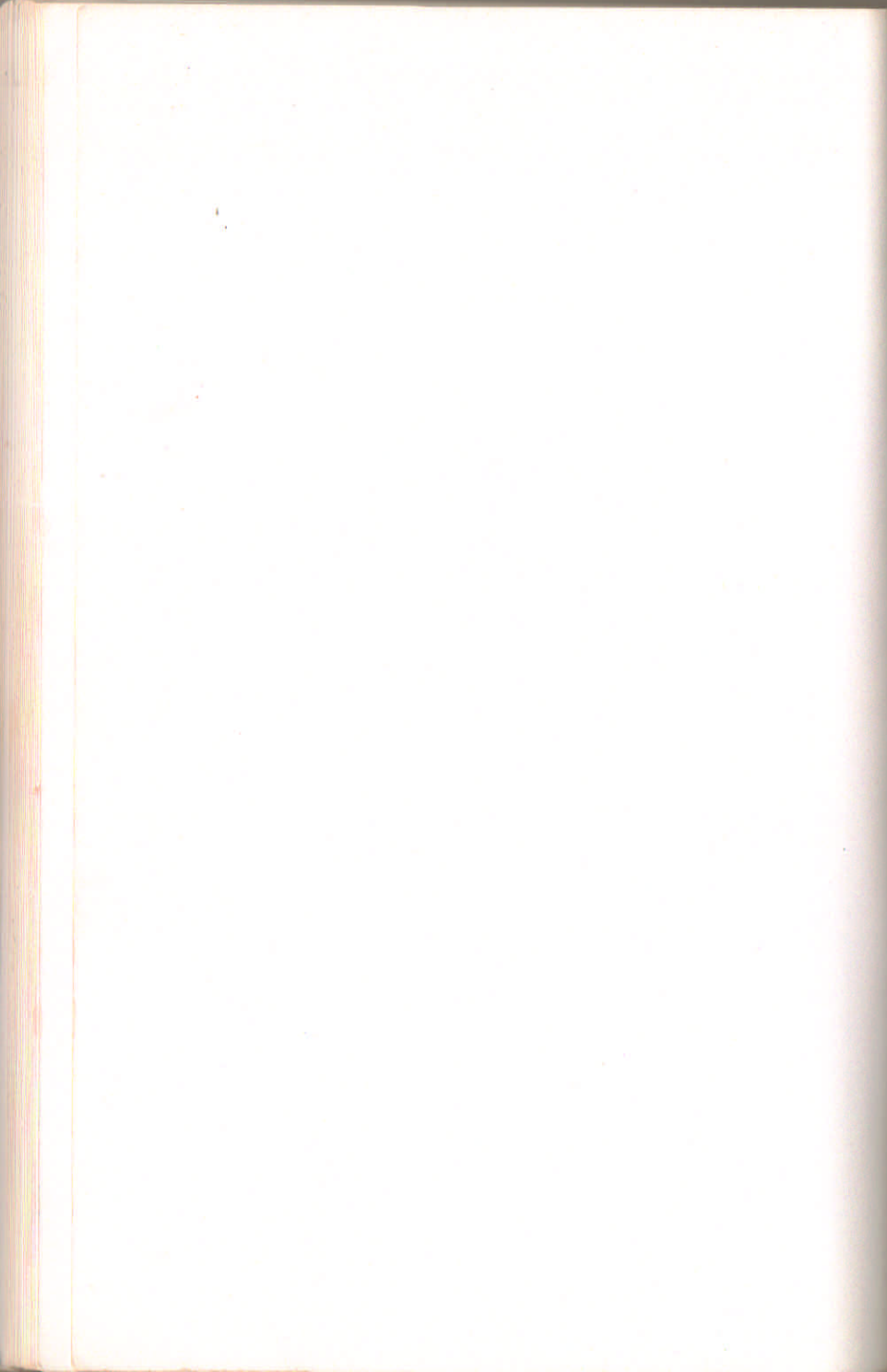
Frank Dacosta. Tab Books

This is a most useful guide to writing adventures. It was the book I first cut my teeth on and it is still proving invaluable now.

**Write Your Own Adventure Programs  
for Your Microcomputer**

Tyler/Howarth. Osborne

At just under £2 this must be the best value adventure book around. Do not be put off by the childish presentation, for the book has some serious things to say.





**THE VIRGIN COMPUTER SERIES**  
Series editors: Tim Hartnell & Clive Gifford

# **ADVENTURES FOR YOUR ZX SPECTRUM**

## **HOURS AND HOURS OF MIND-STRETCHING ADVENTURE GAMES**

Here is your key to a thousand imaginary worlds. Each program, complete within itself, contains a unique set of enemies to be vanquished, environments to be mapped, and problems to be solved. Battle fierce monsters, search for fabulous treasures and weave amazing magic spells – discover all this and much more.

These ready to run Adventure programs – the latest and most popular form of computer games entertainment – have been designed so that when you enter and play them on your ZX Spectrum, you will learn a great deal about writing your own Adventure games. The book includes a detailed explanation of how you can create your own completely original Adventures; and there is a reading list to help you develop your writing and playing skills even further.

**ISBN 0 86369 060 2**

**United Kingdom £2.95**  
**Australia \$9.95 (recommended)**